DTIC FILE COPY

Contract MDA972-88-C-0047

1

DARPA INITIATIVE IN CONCURRENT ENGINEERING (DICE)

AD-A226 394

FINAL TECHNICAL REPORT

PHASE 2
(May 15 1989 - February 15, 1990)

Prepared by:
Kamar J. Singh
Program Manager

Sponsored by:
Defense Advanced Research Projects Agency
Defense Sciences Office
DARPA Initiative in Concurrent Engineering
ARPA Order No. 6511
Issued by DARPA/CMO under Contract No. MDA972-88-C-0047

GE Aircraft Engines
Concurrent Engineering Programs
Cincinnati, Ohio 45215-6301

Contract MDA972-88-C-0047

DARPA INITIATIVE IN CONCURRENT ENGINEERING (DICE)

FINAL TECHNICAL REPORT

PHASE 2
(May 15 1989 - February 15, 1990)

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION
Unclassified

1b. RESTRICTIVE MARKINGS
None

2a. SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE
----

3. DISTRIBUTION/AVAILABILITY OF REPORT
Unrestricted

4. PERFORMING ORGANIZATION REPORT NUMBER(S)

5. MONITORING ORGANIZATION REPORT NUMBER(S)

6a. NAME OF PERFORMING ORGANIZATION
GE Aircraft Engines

6b. OFFICE SYMBOL *(If applicable)*
GEAE

7a. NAME OF MONITORING ORGANIZATION
Defense Advanced Research Projects Agency

6c. ADDRESS *(City, State, and ZIP Code)*
I Neumann Way
Cincinnati, Ohio 45215-6303

7b. ADDRESS *(City, State, and ZIP Code)*
1400 Wilson Blvd.
Arlington, VA 22209-2308

8a. NAME OF FUNDING/SPONSORING ORGANIZATION

8b. OFFICE SYMBOL *(If applicable)*
DARPA

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
MDA972-88-0047

8c. ADDRESS *(City, State, and ZIP Code)*

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO.

11. TITLE *(Include Security Classification)*
DARPA Initiative in Concurrent Engineering (DICE) - Phase 2 (Unclassified)

12. PERSONAL AUTHOR(S)
Singh, Kamar J.

13a. TYPE OF REPORT
Final

13b. TIME COVERED
FROM 1989, 05,15 TO 1990, 02,15

14. DATE OF REPORT *(Year, Month, Day)*
1990, 07,31

15. PAGE COUNT
357

16. SUPPLEMENTARY NOTATION
Prepared in cooperation with: GE-CRD, West Virginia University, Carnegie Mellon University, Resselaer Polytechnic Institute, NC State University,

17. COSATI CODES

FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
Concurrent Engineering, GE, Computer Architecture, CE Research Center, CALS, TQM

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

Abstract:

The mission of the DICE program is to create a Concurrent Engineering Environment that will result in reduced time to market, improved total quality and lower cost for products or systems developed and supported by large organizations. This environment will enable all disciplines important in life cycle of a product or system to cooperate interactively in its definition, planning, design, manufacture, maintainence, refinement and retirement from service.

The DICE Concurrent Engineering environment will emulate for large organizations the human tiger-team approach to concurrent engineering successfully employed by

20. DISTRIBUTION AVAILABILITY OF ABSTRACT
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION
Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL
Larry Pendleton, Jr.

22b. TELEPHONE *(Include Area Code)*
202-694-1771

22c. OFFICE SYMBOL
DARPA

DD Form 1473, JUN 86

Previous editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

16. SUPPLEMENTARY NOTATION (Continued)

Cimflex Teknowledge, Textron Specialty Materials, Saphikon, Inc., Stanford University.

19. ABSTRACT (Continued)

small groups.

This environment will leverage prior experience; it will emphasize early, high-quality decisions; and it will support the propagation of requirements, the feedback of constraints, and the efficient management of risk and change as the product or system is progressively refined from concept to realization.

The DICE mission is to provide an open, computer-assisted environment that implements this concurrent engineering vision. The DICE environment will consist of i) a shared information model that captures complete descriptions of the product or system and all associated process activities and organizational resources, ii) a global object framework that enables the use of the shared information model by a network of cooperating, computer-based clients; and iii) services, methods, tools and advisors that assist concept evaluation, analysis and intelligent decision making.

The DICE mission encompasses a series of rapidly-prototyped demonstrations involving high-valued, complex products in multiple domains and focussing research and development on the validation of the design, performance and scalability of the evolving DICE environment.

The DICE mission will be fulfilled through the establishment of a Concurrent Engineering Research Center to promote education, training, and research in concurrent engineering, to facilitate the transfer of DICE technology to U.S. industrial users and suppliers, and to support the implementation of national initiatives related to concurrent engineering.

The mission of DICE program is to create a Concurrent Engineering environment that will result in reduced time-to-market, improved quality and lower costs for products or systems developed and supported by large organizations. The environment will enable all disciplines important to the life cycle or product or system to cooperate interactively in its definition, planning, design, manufacture, maintenance, refinement, and retirement from services. The DICE environment will emulate for large organizations the human "tiger-team" approach to concurrent engineering successfully employed by small groups.

INDEX

PAGE

1.0 Program Mission and Objectives

The mission of the DICE program is to create a Concurrent Engineering environment that will result in reduced time to market, improved total quality and lower cost for products or systems developed and supported by large organizations. This environment will enable all disciplines important in life cycle of a product or system to cooperate interactively in its definition, planning, design, manufacture, maintenance, refinement and retirement from service. This

~~The DICE Concurrent Engineering~~ environment will emulate for large organizations the human tiger-team approach to concurrent engineering successfully employed by small groups. This environment will leverage prior experience; it will emphasize early, high-quality decisions; and it will support the propagation of requirements, the feedback of constraints, and the efficient management of risk and change as the product or system is progressively refined from concept to realization.

The DICE mission is to provide an open, computer-assisted environment that implements this concurrent engineering vision. The DICE environment will consist of:

- a shared information model that captures complete descriptions of the product or system and all associated process activities and organizational resources;
- a global object framework that enables the use of the shared information model by a network of cooperating, computer-based clients; and
- services, methods, tools and advisors that assist concept evaluation, analysis and intelligent decision making.

The DICE mission encompasses a series of rapidly-prototyped demonstrations involving high-valued, complex products in multiple domains and focussing research and development on the validation of the design, performance and scalability of the evolving DICE environment.

The DICE mission will be fulfilled through the establishment of a Concurrent Engineering Research Center to promote education, training, and research in concurrent engineering, to facilitate the transfer of DICE technology to U.S. industrial users and suppliers, and to support the implementation of national initiatives related to concurrent engineering.

Concurrent Engineering is a revolutionary approach to simultaneously conduct research and development, design, and manufacturing of components and assemblies resulting in a relatively short introduction time for new and advanced high technology materials and processes. It will be possible to include the downstream constraints and requirements at the conceptual stages. By exploiting the parallel

processing methodology now common in the field of computer science, concurrent engineering promises to reduce the introduction time for advanced systems from concept to actual production by one-third to one-half.

The DICE program is the first attempt of its kind to make available concurrent engineering technology to simultaneously conduct research and development, design, and manufacturing in the areas of structural components and electronics. This novel approach will permit examination of multiple options quickly to achieve optimum design and provide active updates to keep all disciplines aware of any changes/modifications. Consequently, options to make changes will be left open to a much later stage in the design cycle.

The DICE program will provide Architecture; Methods, Tools and Advisors; the Manufacturing Demonstrations; a Concurrent Engineering Research Center; and means to transfer technology to industry to achieve the benefits of concurrent engineering.

To achieve the goals for concurrent engineering, a university/industry consortium has been formed to research concurrent engineering issues. This consortium, under the program management of GE Aircraft Engines (GEAE), will demonstrate and validate concurrent engineering tools for:

- mechanical structures
- advanced electronic assemblies

Within the scope of the program, a Concurrent Engineering Research Center (CERC) will be established at West Virginia University, Morgantown, WV. The research results will be "showcased" at CERC and transferred to the industry through a comprehensive technology transfer programs. CERC will conduct symposia and hold workshops in concurrent engineering to dissipate the technology developed/integrated under the DICE program.

Phase 2 of the DICE program commenced on May 15, 1989 and finished on February 15, 1990 for a total of 9 months.

The following participants were involved in Phase 2:

•	GE Aircraft Engines (Prime)	GEAE
•	West Virginia University	WVU
•	Cimflex Teknowledge Corp.	Cimflex
•	GE Corporate Research and Development	GE-CRD
•	Carnegie Mellon University	CMU
•	Howmet Corporation	Howmet
•	Rensselaer Polytechnic Institute	RPI
•	North Carolina State University	NCSU
•	Stanford University	SU
•	Saphikon Inc.	Saphikon

2.0 Summary of Tasks

The four tasks being pursued under the DICE program are:

- **Task 1 -** Concurrent Engineering Research Center (CERC)

  A Research Center is being established at West Virginia University to teach/train the discipline of concurrent engineering. It will be a "showcase" of CE technology for the nation. The research carried out under the DICE program will be integrated, validated and demonstrated at CERC for further transfer to the U.S. industry.

- **Task 2 -** Architecture for CE

  A generic computer architecture will be developed for design and manufacture of both structural components and electronic assemblies. It will be an innovative architecture based on parallel information/computing concepts incorporating existing (both VMS and UNIX) software and hardware "point solutions" in a manner that is relatively transparent to end users.

- **Task 3 -** Methods, Tools & Advisors for CE

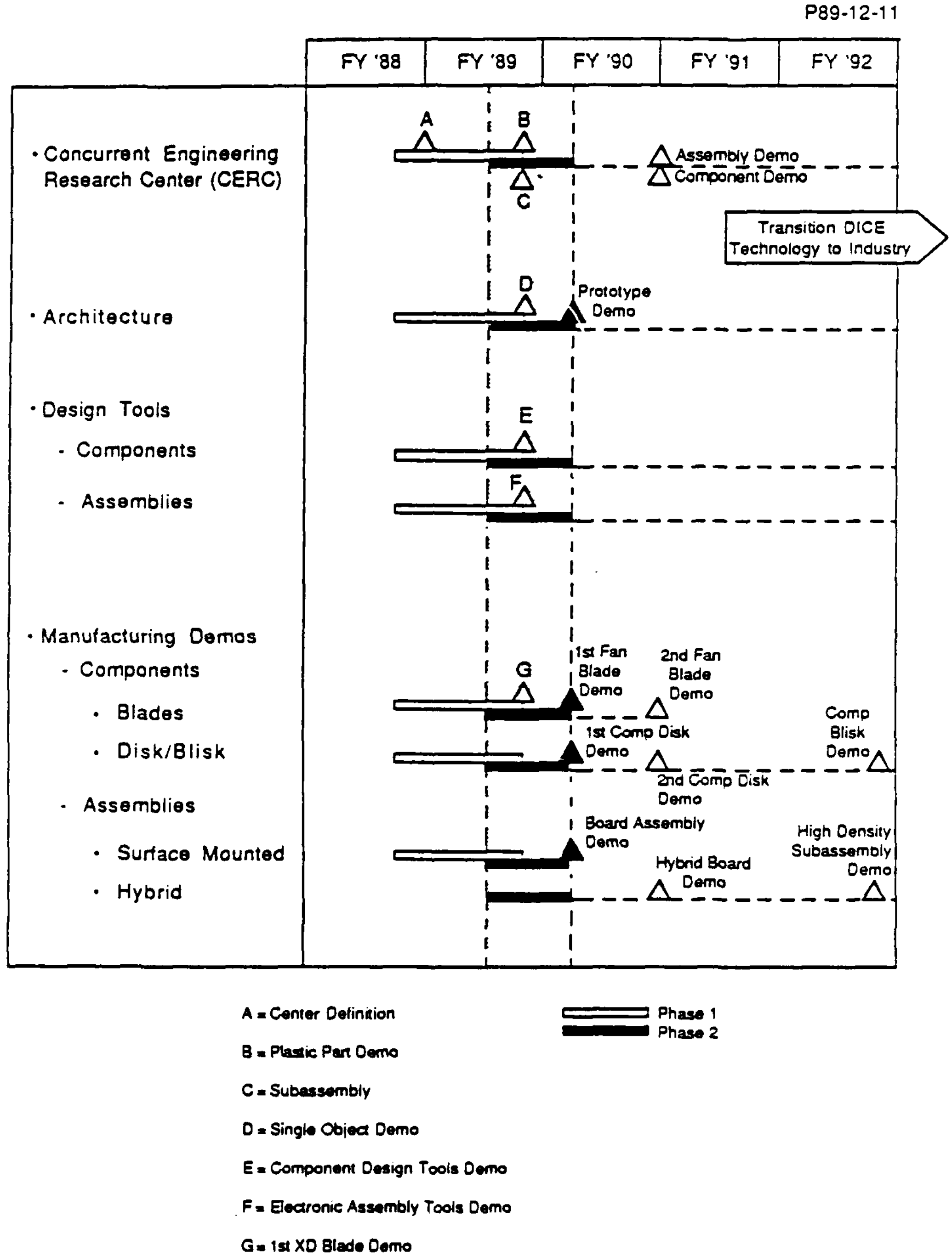
  New design methods, tools and advisors will be developed and along with those that already exist will then be integrated for both structural components and electronic assemblies. These tools will be domain-dependent having "plug in" option with the architecture. As a demonstration, design tools will be developed for new advanced materials and advanced electronic components.

- **Task 4 - Demonstrations of CE**

  Concurrent Engineering technology will be demonstrated by manufacturing structural components from advanced materials and electronic assemblies utilizing DICE architecture and design tools developed/integrated in the program. These demonstrations will move from simple to complex parts as the architecture and design tools become more sophisticated during this program.

A detailed Work Breakdown Structure of the program is shown in Section 3.1.

DICE Program Schedule



3.0 Technical Accomplishments

This section contains the description of technical progress accomplished during Phase 2 of the DICE program, sequenced according to the WBS numbers.

Since there are a number of participants involved in the program and their effort is not integrated at this phase of the program, this report is compiled according to the WBS numbers rather than according to technical effort. However, all the technologies are integrated at the time of the annual demonstration for the customer (Phase 2 demo was held on December 15, 1989). These demonstrations are a means of testing and validating the computer architecture and methods being developed under the DICE program and are used to show reviewable progress.

PHASE 2
Work Breakdown Structure

4.2 CONCURRENT ENGINEERING RESEARCH CENTER

4.2.1.1	Capital Equipment	
4.2.1.1	Computing Environment	WVU
4.2.1.2	Assembly Cell	Cimflex
4.2.1.3	Materials Characterization Lab	WVU
4.2.2	Research Coord. & Administration	WVU
4.2.3	Curriculum	WVU
4.2.4	Faculty/Graduate Students	WVU
4.2.5	------	
4.2.6	------	
4.2.7	Prototyping Facility	WVU
4.2.8	CERC Building	WVU

4.3 ARCHITECTURE (Task 2):

4.3.1	Information Content and Flow	GEAE:
4.3.2	Data Representation	
4.3.2.1	Design Knowledge Representation	CMU
4.3.2.2	Constraint-Based Models	CMU
4.3.2.3	------	
4.3.2.4	------	
4.3.2.5	I-Bus Architecture	WVU
4.3.2.6	DICE to Relational Database Link	RPI
4.3.2.7	Intelligent File Cache	RPI
4.3.3	Information Architecture Prototype	
4.3.3.1	VMS Thread	CRD
4.3.3.2	UNIX Thread	CRD
4.3.3.3	Workstation Node Prototype	CRD
4.3.3.4	Fileserver Node	CRD
4.3.3.5	Info. Management Node	CRD
4.3.3.6	Operating-System-Indep. Thread	CRD
4.3.3.7	PaLS	NCSU
4.3.3.8	Local Architecture	CMU
4.3.3.9	------	
4.3.3.10	Knowledge Server	WVU

4.3.3.11	Information Modeling	WVU
4.3.3.12	Database and PPO Modeling	CRD
4.3.3.13	System Tools	CRD
4.3.3.14	Architecture Coordination	CRD
4.3.3.15	PDES Modeling	CRD
4.3.4	------	
4.3.5	Integration and Implementation	
4.3.5.1	Integration at CERC	CRD
4.3.5.2	Integration at GEAE	CRD
4.3.5.3	Integration of PaLS	NCSU
4.3.5.4	User Interfaces & Graphics Systems	WVU
4.3.5.5	------	
4.3.5.6	Architecture Integration	Cimflex
4.3.6	Validation	CRD
4.3.7	Implementation Language	WVU
4.3.8	Mini-DICE Test Bench	WVU
4.3.9	Soft Prototyping	WVU

4.4 **DESIGN TOOLS**

4.4.1	Enhanced CAD Tools	
4.4.1.1	Process Planning Advisor	WVU
4.4.1.2	Geometric Modeling	WVU
4.4.2	Physical Models	
4.4.2.1	------	
4.4.2.2	------	
4.4.2.3	------	
4.4.2.4	Physical Models	GEAE
4.4.2.5	Database Architecture	GEAE
4.4.3	Macro Models	GEAE
4.4.4	Microscopic/Micromechanics Models	
4.4.4.1	-------	
4.4.4.2	Engineering Models	WVU
4.4.5	Material Properties	
4.4.5.1	XD Material	Completed
4.4.5.2	ICPD Material	GEAE CRD
4.4.5.3	Material Property Reasoning	CMU

4.4.5.4	Material Characterization	WVU
4.4.6	Design Rules	GEAE
4.4.7	Design for Assembly	
4.4.7.1	High-Density PCB	Cimflex
4.4.7.2	Design for Assembly Advisor	WVU
4.4.8	Cost Models	
4.4.8.1	------	
4.4.8.2	Cost Modeler	WVU
4.4.9	Optimization Methods	
4.4.9.1	"Engineous" Software	CRD
4.4.9.2	Optimization of Composites	CRD
4.4.10	------	
4.4.11	Unified Life Cycle Engineering	Stanford
4.5	**MANUFACTURING**	
4.5.1	IPM	GEAE
4.5.2	Manufacturing Facility	GEAE
4.5.3	Process Equipment Design	
4.5.3.1	Process Equipment - ICPD	GEAE
4.5.3.2	Process Equipment - Single Xtal	GEAE
4.5.4	Component Manufacturing Demonstration	
4.5.4.1	------	
4.5.4.2	Plasma Spray Disk	GEAE
4.5.5	Component Testing	GEAE
4.5.6	Assembly Demonstration	
4.5.6.1	Assembly Cell Platform Demo	Cimflex
4.5.6.2	Assembly Cell Prototyping Demo	Cimflex
4.5.7	Assembly Testing	Cimflex
4.5.8	NDE/QA	
4.5.8.1	NDE/QA	GEAE
4.5.8.2	Quality Control Advisor	WVU
4.5.9	Manufacturing Concepts for MMC's	GEAE
4.5.10	Manufacturing Demo - Fan Blade	GEAE

CONCURRENT ENGINEERING RESEARCH CENTER

DICE PROGRAM

TASK 4.2.1.1 Computing Environment — **West Virginia University**

Objectives:

The objective of this task was to design, maintain, purchase and install a heterogeneous computing environment to support the research tasks and the administrative task of the Concurrent Engineering Research Center (CERC) at West Virginia University.

Approach:

Inclusion of hardware and software in the CERC computing environment was determined by:

- User requirements;
- Discussions with other participants in the DICE project;
- Requirements of the software requested;
- Budget restrictions and
- Discussions with vendors.

Technical Results:

In Phase I, CERC created a multi-vendor computing environment. In Phase II, CERC continued to support and enhance the multi-vendor computing environment to:

- support a wide variety of software;
- handle user requirements; and
- continue the development of software that would run on the variety of

workstations and machines found in today's technical computing environment.

In Phase II, the CERC acquired additional space at 2000 Hampton Center. This acquisition required the purchase of additional equipment to support the computing environment at Hampton Center. The major equipment acquired to support Hampton Center included:

- Communications/networking equipment; and
- A disk server.

Conclusions:

The computing environment at CERC was expanded in Phase II with the addition of new equipment. Though the computing environment at CERC was adequate for most users, user needs were not entirely satisfied. Most unsatisfied requests were in the area of software, which could not be obtained due to licensing requirements. The computing environment at CERC needs to expand to meet the growing use of the system. Some of the additional requirements for the computing environment include:

- Software;
- Disk storage;
- Workstations;
- Local disk space on workstations and
- Output devices.

Recommendations:

In the next phase of the project, the computing environment will be enhanced via:

- Acquisition of more workstations;

- Acquisition of more software;
- Addition of more disk space and
- Continued adding of local disks for swapping and paging to the remaining diskless workstations to improve performance.

Publications:

None.

Hardware:

Following are the major hardware items purchased under this task in Phase 2:

- Silicon Graphics (SGI) disk server;
- Five SGI 4D/20 workstations;
- Eight Macintosh computers;
- Five IBM PC computers;
- Three Sun 3 workstations;
- Disk for VAXserver 3600 Ultrix machine;
- Memory for VAXstation 2000 machines;
- Local disks for VAXstation 2000 and VAXstation 3200 machines;
- Two additional disks for Sun 4/280 disk server;
- Memory for Macintosh computers;
- One additional disk for the SGI 4D/240 workstation;
- Additional memory and local disk capacity for 4/110 machine;
- Networking components to provide access to Hampton Center;

- Two Hewlett-Packard pen plotters; and
- Lyon-Lamb real-time scan converter.

The major software items acquired under Phase II include:

- ADDS;
- CEMCAL;
- CLAP;
- COMPAL;
- CYLAN;
- Framemaker;
- Mathematica;
- Microsoft Excel;
- Microsoft Word;
- SAPLAC;
- SAS and
- TURBLADE.

DICE PROGRAM

Task 4.2.1.2 Assembly Cell — Cimflex

1.0 Introduction

This final report is written to document the tasks and results associated with the enhancements and additions to the Phase 1 workcell for high density electronics (HDE) assembly. Cimflex Teknowledge Corporation, located in Franklin, Massachusetts, was assigned the responsibility of making these modifications to further advance the flexibility and performance of the initial workcell. Cimflex was a sub-contractor during DICE Phase 2 to General Electric Company's Aircraft Engine Department located in Evendale, Ohio.

The period of performance of this contract was initially from August 1, 1989, to February 15, 1989. The contract was later extended to March 15, 1990. The workcell currently resides in Cimflex's Pittsburgh facility where it is undergoing further testing and refinement. The workcell is expected to ship to West Virginia University's Concurrent Engineering Research Center (CERC) in the near future when the necessary electrical and pneumatic lines are in place. At that time, installation and final user training will commence. In the meantime, members of CERC's technical staff have been attending briefing sessions at Cimflex's Pittsburgh facility to aid in their understanding of the workcell and its operation.

2.0 Overview and Goals of the Advanced Workcell Initiative

The accelerating trend towards greater component miniaturization, as exhibited by high density electronic (HDE) packaging, is rapidly advancing the capability and power of commercial and military electronics systems. Manufacturers of the complex electronic products that plan to use HDE devices face significant challenges in circuit design and production, where new methods and processes will have to be developed for HDE to reach its final product form.

While the manufacturing technology base must be upgraded to handle the latest advances in electronics technology, the need to provide a greater variety of end products, shorter development lead times, and lower unit costs remain as high operational objective. Defense and aerospace suppliers are confronted with the most stringent of these requirements, and must simultaneously satisfy exacting standards for product quality, reliability and environmental stability. The combined effect of these factors results in greater costs and development cycles for weapon systems used today, and promises to increase as HDE and other technologies come to the forefront. Much of the development cost and time is associated with the iterative design, prototyping and testing of new printed wiring board assemblies (PWAs) and high density modules.

The mission of this Assembly Workcell Project is to develop fundamental manufacturing system technology to enable rapid prototyping and testing of current

and emerging HDE circuits. The primary goal is to minimize the iterative circuit and process development times by providing the capability for new designs to be assembled and tested within 24 hours. This capability is expected to provide significant benefits to the defense electronics industry where:

- Development and engineering change order (ECO) cycles are traditionally long, and in many cases are inadequately monitored and controlled.
- Small lot sizes require frequent change-over of parts and processes.
- Current assembly methods are labor intensive and require extensive training.
- Quality suffers from inconsistent manual assembly and errors in judgment.
- Components are extremely expensive and in short supply.
- Documentation is extensive and involves excessive handling and distribution.
- Delays in test data feedback preclude timely redesign.
- Process development requires multiple set-ups and trials.
- Repeated handling results in breakage or circuit failure.
- "Build short" is a common practice to optimize labor utilization.
- Prototype build is not representative of actual production.
- Rapid order filling is essential to National security.

Much of the Defense Electronics community is caught up in the demanding test and approval processes for new designs. Production implementation of advanced HDE technologies lags the commercial sector. The transition from lead-through-hole technology (LTH) to surface mount devices (SMD) to fine pitch devices (FPD) will be slow and will result in a need to perform a wide variety of assembly operations on a flexible basis in the coming years. Integrated systems comprised of automated assembly workcells are required to meet this challenge while improving the cost, quality and lead time for Defense electronics.

The system requirements to facilitate rapid prototyping of HDE modules involve a broad range of functional parameters, technical disciplines and manufacturing logistics. A primary requirement is that the manufacturing system be a responsive, active part of the overall production environment. This implies the assembly workcells must be data driven via upstream Concurrent Engineering Design Workstations (CEWS) and other management systems. Also, the assembly workcells must establish generic standards for product design, manufacturing engineering, etc., to rapidly deploy new designs in a consistent and manageable fashion. These standards should be reflected in a workcell design which allows simple configuration of the assembly production line (microfactory) as well as quick reconfiguration of manipulators, parts feeders and tools to react to different process and product requirements. Also, the workcells must be founded on a platform capable of supporting current and emerging HDE technologies to avoid early obsolescence. Key system attributes of these Assembly Workcells are outlined on the next page.

Key Attributes

- Communication Interface and Software Links to enable/support:
  - factory floor networking to CE workstations
  - off-line workcell program generation (concurrent with production)
  - automated, flexible work order entry
  - CAD data transmission
  - real time production queue and performance data monitoring
  - current workcell mode, status and alarm reporting

- CAD Download Interface Software to enable/support:
  - rapid, automatic "self teaching" of component placement points
  - a variety of commercially available CAD workstations
  - automatic data interpretation/resolution of work order parameters.
  - automatic data validity check and completeness against work order

- Database Software Environment to enable/support:
  - rapid assembly program development for new PWBs/ECOs
  - CAD data logging and point file organization
  - configuration libraries for a variety of feeders, tools, and process parameters
  - configuration libraries for a multitude of components

- Generic Software to enable/support:
  - rapid workcell changeover (and *verification*) of feeders and tooling
  - simplified, task based subroutines to ease programming
  - automatic work order conversion to workcell operational formats
  - automatic designation of setup parameters and tool assignments
  - a high level language for simplified operator training/interaction

- Enhanced Screenware to enable/support:
  - rapid, menu-driven setup and changeover of feeders and tooling
  - icon based display of workcell mode and status
  - interactive query-based system for production data acquisition/reporting

- Integrated Grayscale Vision to enable/support:
  - Inspection of component leads for quality and orientation
  - Inspection of PWB fiducials for spatial registration
  - multiple cameras to accommodate a high mix of job specific tasks

- Inherently Accurate Manipulators to enable/support:
  - CAD driven placement of fine pitch SMDs
  - predictable response to vision guidance
  - stable and consistent repeatability without recalibration job to job
  - quick change grippers and tools for flexible production

- A powerful controller/system architecture to enable/support:
  - multi-tasking of drives, I/O, sensory inputs and communications
  - simultaneous off-line program development
  - high speed adaptive control of manipulators and tooling
  - flexible, modular incorporation of optional/new workcell devices
  - large working memory and data storage for high mix production

- An Integrated Generic Workcell Platform to enable/support:
  - rapid adjustment of the substrate transport/index system
  - built-in transport buffers to facilitate "board lot size of one" manufacturing
  - removable/changeable manipulators for low downtime/quick change
  - a stable, serviceable foundation for consistent performance/high yield
  - integration of emerging technologies and new production processes
  - production method standardization for concurrent engineering
  - flexible adaptation to a variety of process tasks
  - modular implementation of a production line or microfactory
  - rapid reconfiguration of workcells/lines for changing demands

- HDE Assembly Technologies Application to enable/support:
  - feed trays for various FPD packages
  - quick-change, intelligent component feeders and manipulator tools
  - programmable dispensing of adhesives, solder paste, flux, etc.
  - programmable, pressure controlled placement of FPDs for high yield
  - flexible vision inspection of leads for a wide range of FPDs
  - vision guidance at placement site for improved accuracy
  - generic FPD application software for rapid programming and setup

The system requirements above illustrate the individual functional items and features necessary to meet the overall mission objectives. Some of these items and technologies exist today in various forms and application environments, while others need to be developed. The proposed program seeks to specify, develop and integrate these modular elements into a cohesive, organized system for HDE Assembly Rapid Prototyping. The DICE Assembly Workcell Project was launched under this premise and significant progress has been made toward the initial objective to develop a first level prototype workcell with many of the features outlined above. The Phase 2 program allows significant functional enhancements and improved prototyping flexibility of the first demonstration workcell. This enhanced workcell will form the essential beginning of a subsequent, totally integrated rapid prototyping facility for HDE assemblies.

The following sections review the basic technologies included in the first workcell and describe the enhancements and associated technology development for Phase 2.

3.0 Summary of DICE Phase 1

The focus of the Phase 1 assembly workcell was to develop and demonstrate a generic, high performance applications platform for rapid prototyping of high density electronics assemblies. The key aspects of this platform are:

1.) An advanced manipulator system based on Sawyer Motor Technology to achieve high accuracy placement of components.

2.) An integrated vision system for manipulator guidance and component inspection.

3.) A database programming environment for improved application development.

4.) A CAD interface to allow off-line data transfer between the workcell and Concurrent Engineering Workstations (CEWS).

The demonstrated application for the Phase 1 workcell was the placement of high lead count, fine pitch surface mounted devices. The manipulator platform was configured with matrix trays to present components, a vacuum pickup tool to retrieve components out of the trays, and an in-line circuit board transport module to convey PWBs into (and out of) the workcell envelope for assembly. The vision system provided the capability of inspecting the quality of component leads prior to placement as well as accurately directing the manipulator heads after detecting the alignment of PWBs and components. This application validated the ability of the workcell platform to not only place components with extreme precision, but also to operate in a software driven, user-friendly mode. (See Exhibit 1 for a picture of the workcell.)

4.0 Phase 2 Overview

The intent of enhancing the Phase 1 workcell was to improve its flexibility and performance to effectively respond to the introduction of new and constantly changing designs typical of actual rapid prototyping environments. For example, providing increased capacity to place a variety of standard SMD components, as well as fine-pitch devices, expands the assembly domain to include a wider range of HDE prototypes. The integration of solder paste dispensing prior to placement greatly improves system flexibility and process yield by eliminating an independent, upstream operation. The addition of modular, intelligent feeders and manipulator tools allows rapid conversion of the workcell in support of rapid prototyping. First pass yields increase with the use of new vision technologies, such as pin-to-pad matching for automatic component registration. Combined with an integrated database and applications software, the features described above make up the basis for the enhanced workcell in Phase 2.

EXHIBIT 1

4.1 Develop and Install Enhancements to Phase 1 Workcell

4.1.1 Anti-Collision Software

One of the unique features of the Sawyer motor technology based system is the ability to have multiple manipulators running in the same work envelope to share feeders, tools, and assembly tasks. Accordingly, anti-collision software is necessary to prevent the manipulators from crashing into each other. A user definable anti-collision strategy has been developed which permits N number of manipulators to operate simultaneously over the same area. When an impending collision cannot be resolved, the manipulators stop moving and a message is displayed on the screen informing the operator that intervention is required. With proper programming, the occurrence of a gridlock situation can be greatly minimized if not eliminated altogether.

4.1.2 Modular, Intelligent Taped Reel Feeders for Conventional SMDs

Standard SMDs are usually packaged in compartments on taped reels of various widths. Modularity is accomplished through the use of common electrical and mechanical interfaces. Without this capability, large sections of the workcell envelope would be dedicated to certain feeder types, and thus limit the range of feeders that could be used. Lastly, these feeders need on-board sensors to recognize that a part has been retrieved so they can index the next part into place. Four (4) of these reel feeders of different widths are provided: 16mm, 24mm, 32mm, and 44mm.

These feeders were purchased as off-the-shelf items and were integrated onto the feeder table with common mechanical interfaces. An electrical interface box was provided which can supply power for up to 20 feeders. (See Exhibit 2.)

4.1.3 Pin-to-Pad Matching Vision Capability[1]

Most vision guided assembly systems today rely on a ballistic approach to component placement. That is, they inspect the circuit board for its orientation and separately inspect the component for its position, and then place the part onto the board. Used in this way, vision compensates for component and board presentation inaccuracy, but does nothing to compensate for manipulator inaccuracy. This type of approach has proven successful for placing components having lead pitches down to 0.025 inch, for which the placement accuracy is typically specified as 0.003 inch. The main factor limiting the placement accuracy which can be achieved using this type of approach is usually the accuracy of the manipulator. Even though the components and boards are located using vision, the manipulator must still make an accurate move to put the two together. The final placement accuracy will be no better than the accuracy of this move.

[1] Major portions of this section were provided by Niall O'Driscoll from Adept Technologies

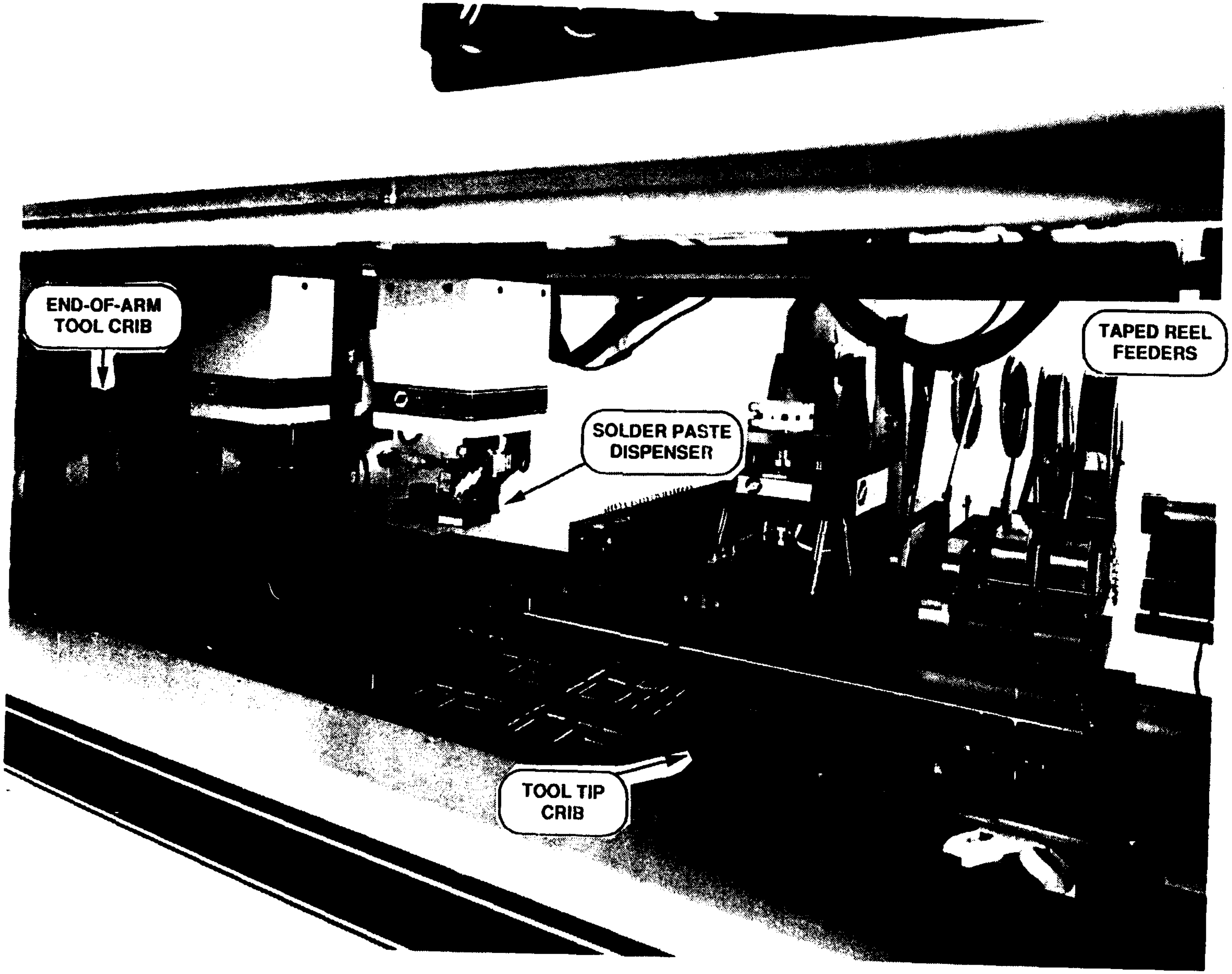

END-OF-ARM
TOOL CRIB
SOLDER PASTE
DISPENSER
TAPED REEL
FEEDERS
TOOL TIP
CRIB

EXHIBIT 2

To improve placement accuracy, therefore, it seems reasonable to start by improving manipulator accuracy. As accuracy goes below 0.003 inch, however, manipulator cost rises dramatically, payload drops, workspace drops, and flexibility drops. In addition, it becomes necessary to control environmental factors such as temperature and humidity. Field replacement of critical components such as encoders or links entails time-consuming recalibration procedures. In short, extremely accurate manipulators are extremely expensive and difficult to build and maintain, especially if they are required to maintain their accuracy over a broad range of operating temperatures and payloads.

The accuracy of any manipulator will ultimately be limited by its resolution, but high resolution can be achieved much more easily than high accuracy. High resolution requires only the installation of highly sensitive position sensors. To improve accuracy requires maintaining of much tighter machining and assembly tolerances for all components which make up the kinematic chain between the manipulator base and tool tip, or development of complex software models to measure and compensate for differences between an ideal manipulator and a manipulator as it is actually built. The SMD placement technique using the Pin-to-Pad matching gripper avoids many of the problems and costs associated with manipulators having inherently high accuracy, and relies instead on manipulator resolution. The X-Y axis positioning resolution of the DICE workcell is less than 0.0005 inch.

The reason that conventional approaches to SMD placement require manipulator accuracy is that during placement they make no direct measurement of the position of the component relative to the board artwork. Instead, this position is calculated. If the position of the component relative to the board artwork could be measured directly during placement, the need for manipulator accuracy would be greatly reduced. If the component-to-board misalignment is measured directly just prior to placement, the information can be used to command the manipulator to make a small incremental move to improve the alignment. This measurement can then be repeated, and additional movements commanded, until the desired degree of accuracy is achieved. The implementation of such a placement scheme using the DICE workcell equipped with an AdeptVision XGS vision system and an end-effector specially designed for this purpose.

<u>End-effector:</u>
Dual camera end-effector with vacuum tip. Cameras are on opposite sides of vacuum tip, with center to center spacing equal to the distance between diagonally opposite corners of the component. Cameras each have 0.5 inch field of view. Vacuum tip has 0.25 inch pneumatically actuated z-stroke which raises and lowers it independently of the cameras and end-effector.

A drawback in the current design of the end-effector is the fixed position of the two cameras. Although the cameras are mounted on a slotted bracket, manual

intervention is still required to adjust the spacing of the cameras when drastically different component sizes are used. Otherwise, the parts will not be in the field of view. Moreover, the smallest component that can be viewed is predetermined by the center-to-center distance of the cameras when they are brought together with no space between them.

Process Overview

In transferring a component from a feeder to the board, the manipulator performs the following steps.

1) Acquires the component from the feeder, gripping it so that each of the two end-effector cameras has a view of a diagonally opposite corner of the component.

2) With the vacuum tip in the down position, measures the positions of the leads on each of the two corners using images acquired from each of the two cameras.

3) Pneumatically raises the vacuum tip.

4) Moves over the component placement site.

5) Registers each of the two local fiducials using the end-effector cameras, and calculates their positions relative to the previously determined lead positions.

6) Computes the correction required in X, Y, and rotation about Z necessary to bring the leads into proper alignment with the local fiducial.

7) If the measured misalignment is unacceptable, makes a move to reduce it.

8) Repeats steps 5 through 7 until any misalignment has been reduced to an acceptably small level.

9) Pneumatically lowers the component into position and releases it.

The pneumatic Z-actuation of the vacuum tip is required to prevent the component from contacting the board while the final position adjustments are made. When registering the board artwork, it is essential that the cameras be positioned the same distance from the board that they were from the component leads when the leads were registered. If the component could not be raised independently of the cameras, this requirement would put the component in contact with the board when the fiducials were registered. Clearly, it is important that the end-effector be designed so that when the component is lowered for placement it returns to its original position.

Component Requirements
For the approach described above to work, it is essential that the component leads and board artwork be viewed with the same cameras. This requirement makes this approach most suitable for components having leads which can be viewed from above, such as gull wing or TAB components. J-leaded components are not suitable because the only part of the lead which can be seen from above is the shoulder. The fact that J-leaded devices cannot be viewed using the tool is not an issue because these devices typically have lead pitches of 0.050 inch and do not require the same level of positioning accuracy as fine pitch parts (i.e., less than 0.025 inch lead pitch).

Board Requirements
The difficulty of obtaining an image of the pads when the leads are sitting directly above them, or when they are otherwise obscured by solder paste etc., makes it necessary to have local fiducial at each placement site. These fiducials must be positioned in the corners of the placement site corresponding to the corners of the component which were registered by the end-effector cameras. They need not be in any particular location in the field of view, but their ideal locations relative to the component corners must be known either from CAD data or some other source. This requirement implies that the PWB design engineers have prior knowledge of this producibility guideline for the technique to be used. Fortunately, the addition of corner fiducials for each fine pitch part is a relatively trivial design consideration.

Vision Algorithms
The field of view of each camera is approximately 0.5 inch square, and each camera has a resolution of 484 by 509 pixels, providing a resolution of roughly 0.001 inch per pixel. To achieve placement accuracy approaching 0.001 inch, therefore, would require that sub pixel measurements be made of the lead and fiducial locations. For this reason, measurements of the lead positions are averaged over many leads. An additional justification for measuring many leads is that the contribution of any one slightly bent lead is thereby minimized.

Once the corners of the component are registered, the component is raised relative to the cameras and transferred to the board. With the end-effector in position above the board, the local fiducials are registered. These fiducials are simple circles, and were located using a vision tool called arc fitter. This tool provides more accurate registration of circular regions than is provided by simple centroid calculations, and gives results accurate to better than 0.1 pixels.

Knowing the locations of two corners of the component relative to the local fiducial, and knowing from CAD data the desired locations, the positional error in X, Y, and rotation about Z can be calculated. The manipulator is moved by this amount, and the fiducial re-registered. The process of measuring and correcting is repeated until the measured error is reduced to less than 0.0007 inch, at which point the component is lowered into position.

Testing was performed on a 256 lead ceramic quad flat pack with 0.020 inch lead pitch. This device was chosen because it was large enough to be viewed by the system and it also represented a fine pitch part. The main drawback to the device was that the leads exited the component body straight out with the tie bar attached. Normal operating conditions would require the tie bar to be removed and the leads formed in a gull wing pattern. The cut and form operation involves an extremely expensive die which was not available to us. The tool was able to be tested nonetheless, and demonstrated the functionality described earlier. Future tests will involve more representative components.

4.1.4 Solder Paste Dispensing Unit

In high volume, low mix SMT manufacturing facilities, a screen printer is used to deposit solder paste onto the substrates before part placement. Within the screen printer is a stencil or silk screen that matches the pattern of the pads on a given circuit board type. Every time a new board is introduced, a new stencil must be made. This approach would be unworkable in a rapid prototyping environment because the cost and time associated with constantly making new stencils would be prohibitive. A better method is to programmatically apply the solder paste with a syringe using one of the manipulators to direct the tip to the right location on the board in conjunction with a precision controlled dispensing system to meter the appropriate amount/size of solder paste deposits.

The dispenser selection process started with an evaluation of an Archimedes Screw (Auger) Valve. This valve forces material down the threads of a screw to the tip or opening of the chamber in an "Auger feed" manner for low pressure dispensing. This technique is very good when dispensing homogeneous (i.e., adhesives) materials and particularly when dispensing continuous lines. Another critical advantage is the ability to automatically control and change the volume dispensed. Also, this valve will generally not clog since the material is "dispensed" in a "pin transfer" or "dabbing" manner. Unfortunately, the volumetric accuracy and repeatability are impaired due to this "dabbing" and unavoidable material leakage around the screw. Material separation and particular breakage due to the "churning" action of the screw are severe limitations to this valve's usefulness. Low viscosity materials cannot be dispensed with this valve.

After conducting additional research, a piston positive displacement pump was chosen instead of the Auger type described above. This valve transfers the material from a syringe holding chamber through a channel to a small reservoir. As the piston strokes upward, the reservoir fill sunder low (typically 5 psi) pressure. The piston then strokes downward, applying pressure to only the small amount of material in the dispensing reservoir. This material is then dispensed very rapidly under high (typically between 5,000 - 10,000 psi) pressure. This valve is very good for dispensing both homogeneous (i.e., adhesive) or nonhomogeneous (i.e., solder paste) materials.

Since the "high" pressure is only applied to the dispensed volume, material separation does not occur. Also, the volumetric accuracy and repeatability are very good, typically 1% - 5%. Unfortunately, the volume dispensed must be mechanically changed and past experiments have indicated that some (1% - 3%) partial breakage occurs. Also this valve can "clog" if not properly utilized.

As each dot of solder paste is dispensed from the pump, it is necessary for the shot to come in contact with the PWB or substrate so that surface tension will pull the solder paste off of the syringe tip and deposit it on the board. This requirement implies that the distance between the nozzle tip and the substrate is known. The technique chosen to accomplish this task was a laser range finder which involves shining a laser beam down onto the deposit site and using a sensor to translate the reflected angle into a height measurement. See Exhibit 2.

Although the solder paste dispensing pump with integrated tip-to-board height detection was assembled and tested in a standalone mode, the unit was not implemented on a manipulator to test for automatic operation. What remains to perform this function is to write the software to coordinate the motion of the manipulator and the dispensing action. Setting up the dispensing parameters for various component types in the database also needs to be completed.

4.1.5 Auto-Change End-of-Arm Tooling and Tool Crib for Various Nozzles

SMDs come in a wide range of sizes and form factors. One vacuum pickup tool is incapable of acquiring all the SMDs from the numerous feeders available. Accordingly, the manipulators must have a means of automatically exchanging nozzles for different part types. This feature is provided on both manipulators for interchangeability purposes. Two tool cribs, one for each manipulator is located within the workcell to store the nozzles. The workcell will software track the location of each tool in the crib. See Exhibit 2.

With the addition of the Pin-to-Pad gripper and the solder paste dispensing tool, an end-of-arm tool (EOAT) crib was developed for the automatic storage and retrieval of these items. Each side of the workcell contains an EOAT crib to support both manipulators. Lastly, the cribs are mounted on slides to move in and out of the work envelope when needed to preserve usable space. Again, see Exhibit 2 for a picture.

4.1.6 Other Enhancements

- Manipulator Packaging - Provided a main manipulator cover to protect operators while the system is in operation.
- Umbilical Cord Replacement - Provided a high flex manipulator cable to reduce drag and improve performance.
- Vision Lighting - Provided new lighting on each manipulator for downlook vision inspection (i.e., fiducials) using LEDs to minimize the effects of ambient lighting.

DICE Program

Task 4.2.1.3 Materials Laboratory **West Virginia University**

Objective:

This project was designed to strengthen the research facilities at West Virginia University for the synthesis and characterization of advanced material. The advanced materials included stable compounds and intermetallics, artificially stabilized structures, composites and long-lived metastable structures. The overall objectives are to develop a research program for designing advanced materials with superior properties (low density, high temperature strength, creep resistance, environmental stability, etc.) so that tailoring the properties of these materials for a variety of applications may become possible.

Approach:

There were two components to this program: experimental and theoretical studies of TiAl alloys and acquisition and development of new experimental facilities for synthesis and characterization of advanced materials. The TiAl alloys were prepared using an existing triarc furnace, and these samples were characterized by a variety of techniques (metallography, x-ray diffraction, thermal expansion, hardness testing, specific heat measurements, SQUID magnetometry). In the theoretical program under Professor Cooper, supercell total energy calculations were used to model the behavior of dilute alloys of g-TiAl doped with transition elements (Mn,V,Cr) with particular attention to crystallographic stability, the calculation of the lattice parameters, bulk modulus Youngs' modulus and site selection of impurities.

Two new pieces of equipment were acquired under Phase II. These included a sputtering unit for the preparation of thin films and multilayer structures of intermetallic systems and an optical microscope (metallograph) for characterizing the grain boundaries, the grain size and different phases of metals and alloys. A list of all pieces of equipment acquired under Phases I and II is given under Hardware.

Technical Results:

Titanium-aluminides are among the most promising ordered intermetallic alloys for high temperature structural applications since they have high melting points, low density, good oxidation resistance and high elastic modulus. However room temperature embrittlement (low ductility) of these alloys has so far prevented their use in structural applications. Recent studies have shown that incorporation of a small amount of a third element (e.g. Mn) into the lattice provides some improvement in room temperature ductility. We have carried out studies of the structural, mechanical, electronic and magnetic properties of Mn doped g-TiAl alloys with nominal Mn concentrations of 0.1, 1, 2 and 5 atomic %. The goal of these studies was to determine how Mn incorporates into the structure, what its electronic configuration is and how the crystal structure is affected. Our results show that with Mn doping, tetragonality of the unit cell decreases and a localized magnetic moment = 2.1 m_B per Mn atom appears. Our x-ray scattering studies show that, for small dopings, Mn favors the Ti sites of the g-TiAl structure, in agreement with recent theoretical estimates using total energy calculations. The improved ductility with Mn doping may in part be due to an increase in the possibility of deformation modes resulting from reduced tetragonality and reduced symmetry.

Conclusions:

Our experimental measurements and theoretical estimates show that Mn doped in g-TiAl alloys tends to occupy the Ti site, has a magnetic moment associated with it and reduces the tetragonality and size of the unit cell. The improved ductility may be due to a combination of these factors.

Recommendations:

Additional experimental and theoretical studies involving other dopants from the transition elements and light elements should be tried to improve the ductility of the titanium aluminides. Since electronic structure and crystallographic properties appear to be strongly correlated, a variety of experimental measurements

combined with theoretical studies are needed to understand the properties of these alloys and to design suitable alloys for desired high temperature applications.

Publications:

1. Coletti, J. (supervised by M. S. Seehra) "Observation of a Localized Moment in Mn Doped g-TiAl Alloys", Thesis, West Virginia University, 1990.

2. Coletti, J., V. Suresh Babu, A. S. Pavlovic, and M. S. Seehra. "Observation of a Localized Moment in Mn Doped g-TiAl Alloys." Under review by CERC/DICE for submission to the Physical Review.

3. Khowash, P. K., D. L. Price, and B. R. Cooper. "Prediction of Site Selection for Additives to Intermetallic Compounds." To be submitted to Physical Review Letters.

Hardware:

1. X-Ray Diffractometer; Rigaku model D/Max, with computer control, and accessories for low temperature and high temperature studies; installed in room B-17 Hodges Hall; Cost = $136,654.

2. SQUID, Model MPMS by Quantum Design Inc., with accessories for high temperature studies and hysteresis measurements; about half ($60,000) of the cost of this instrument were provided by west Virginia University as laboratory setup for Dr. Abdul-Razzaq, one of our new assistant professors; installed in room B-13, Hodges Hall; Cost = $128,050.

3. TMA (Model TMA40 by Mettler Instruments Inc.); with accessories for high temperature (to 1000°C) measurements; installed in room B-16 Hodges Hall; Cost = $42,918.

4. Hardness Tester (Model 940-142); installed in room B-05 Hodges Hall; Cost = $6,343.

5. Rotating Anode System; 18 kW Generator by Rigaku. Four Circle and Goniometers by Huber, Detectors by EG&G and Tennelec, Computers by

Apple, and other accessories to be installed in room G-40 Hodges Hall. This system was assembled with the assistance of Dr. Wilson, our new assistant professor; System cost = $250,000.

6. Dr. N. S. Dalal, Department of Chemistry, has received equipment for the measurements of variable frequency microwave dielectric loss and surface resistivity. The unit has been assembled from various components (10 MHz-20 GHz Synthesized Sweeper, Scalar Network Analyzer, Directional Bridge, AC/DC Detector etc.) purchased from Hewlett Packard. A microwave cavity with a center frequency of 15 GHZ is being fabricated for use with the system. The system is installed in the Chemistry Research Laboratory. System cost = $50,000.

7. Sputtering Unit: A custom-made four-gun system by Cook Vacuum Inc. consisting of facilities for cooling and heating substrate; specially designed for fabrication of multilayer intermetallics; includes a DEK-TAK unit fcr measuring film thickness; Installed in room G-40 Hodges Hall; Cost = $180,000.

8. Versamet Metallograph by Buehler Ltd. with accessories; Installed in Room B-05 Hodges Hall; Cost = $16,616.

DICE Program

Task 4.2.2 CERC Administrative and Technical Services — **West Virginia University**

Objectives:

This task focused on the development of a team-oriented organizational structure for the CERC and its research activities, both within the center and between the center and the other participant companies and institutions in the DICE program consortium.

Approach:

Three basic levels of team group coordination were identified and pursued by the CERC:

1. Coordination among the various research tasks within CERC.
   This is an important level of coordination for the coherence of the CERC team and to assure that the efforts of all its members follow a common, integrated path. Such coordination was implemented through regular staff meetings, scheduled meetings among group leaders involved in different research tasks, review and planning meetings between CERC and the research teams and through an in-house professional development seminar series.

2. Coordination among the various organizations involved in the DICE program.
   This type of coordination was pursued both formally, through the group leaders for each organization and institutions and informally, through individual task leaders who share common interests and objectives and work in five basic research teams: Architecture, Methods, CERC, Demonstration/Integration and Customer Focus.

3. Coordination within DICE Teams.
   Phase 2 of the DICE program facilitated and formalized the interactions among all the program participants in certain technical and administrative areas. This type of coordination was implemented

primarily within two teams -- Architecture and Methods. Most research activities fell into one of four teams and were subsequently incorporated into one of the four teams: Architecture, Methods, CERC or Demonstration/Integration.

Technical Results:

DICE represents itself as a large, remotely distributed development effort involving contributors from many constitutions, both commercial and academic. Several painful, failed attempts to achieve consensus on architecture concepts and other issues led to the reorganization of the program into five interacting teams (Architecture; Methods; Demonstration/Integration; Customer Focus and CERC) whose leaders are members of a governing executive team. Besides program planning and monitoring activities, the Executive Team ensured that customer requirements were identified and incorporated into the technical objectives of the sub-teams. Each of the teams underwent facilitated training to overcome cultural and behavioral impediments to cooperation and to evolve a mission-oriented focus. The Demonstration Team implements and orchestrates technology demonstrations according to the scenarios and application methods, tools and advisors provided by the Methods Team and the capabilities of the information framework and services provided by the Architecture Team. The demonstration scenarios were derived from actual product-development process diagrams captured with the help of industrial partners.

Conclusions/Administrative Actions:

Invitations and "Call for Papers" were issued for the Second National Symposium on Concurrent Engineering to be held at the Lakeview Resort and Conference Center in Morgantown, West Virginia, on February 7-9, 1990.
Nearly twenty-five approved, external technical papers were written in the areas of geometric modeling, utilities and services and architecture. Some forty papers were presented at the Second National Symposium on Concurrent Engineering.
The Concurrent Engineering Research Center leased additional space to house its research effort at 2000 Hampton Center, Morgantown, West Virginia. This new facility also provides approximately 2500 square feet of new space

required to maintain the DICE/CERC Testbed. This satellite facility is located approximately two miles from the CERC and is now occupied.

The CERC established an office for Publications and Program Advancement to promote the program, disseminate public information and facilitate marketing contacts and partnerships, and administer the technical report series and the Library.

The Center staff designed and implemented a new computerized accounting system for Phase 2 to more efficiently and effectively track expenditures and to increase the response time for monthly reporting.

The CERC hired three new full-time staff employees. These included the positions of Operations Manager, Research Associate and an Accountant III.

Task members prepared and submitted a comprehensive Equipment/Furniture List for GE and DARPA outlining invoice number, cost and placement in CERC.

A quarterly newsletter was published and distributed to include many recipients in Academic/Industrial Facilities. Some 1200 names are now included in the CERC mailing list.

The lay-out of space and equipment for the Advanced Prototyping Center was initiated with first cut drafts and reviews provided.

The Center administration:

- Developed a preliminary business plan for CERC;
- Coordinated the Phase 2 Demonstration held at CERC December 15, 1989. These efforts involved all the preparations leading up to the demonstration;
- Coordinated the Second National Symposium on Concurrent Engineering held February 7-9, 1990;
- Developed the duties and responsibilities for the consultants and sub-contractors to be hired in Phase 2. These Statements of Work include work to be performed, place of performance, deliverables, milestones, costs, etc.;
- Reworked the Phase 2 budget proposal for GE and DARPA with changes in time of performance and budget appropriations with accompanying contract revisions, and technical analyses;
- Worked the Phase 3 Technical and Cost Proposals;
- Implemented a comprehensive video-conferencing facility at CERC;

- Facilitated a professional in-house staff development series, including courses in ROSE and QFD;
- Participated in a Team-Building Executive training series resulting in a restructuring of CERC as a team management concept;
- Established a comprehensive CE Library at the Concurrent Engineering Research Center and
- Established an electronic CE abstract and information services called CERCnet.

Recommendations:

This effort does and will continue to work for the advancement of the CERC initiative. The DICE team organization and behavioral norms provide a case study for others who may need to learn that culture and organization are essential to the success of concurrent engineering.

Publications:

Publications during this Phase included:

- Materials to promote the *Second National Symposium on Concurrent Engineering*;
- Proceedings of the Second National Symposium on Concurrent Engineering;
- CERC and CERCnet promotional materials and
- Library bibliographies and catalogues.

Hardware:

Furniture and library equipment were purchased for the National CE Repository.

DICE Program

TASK 4.2.7 - Prototyping Facility **West Virginia University**

Objectives:

The objectives of the task were to develop and establish a modular, scalable, and flexible testing and manufacturing facility to:

(i) evaluate and demonstrate the Concurrent Engineering (CE) Concepts from design to prototyping manufacturing of engineering components;

(ii) accelerate the transfer of CE Technology into industrial production; and

(iii) provide conceptual prototyping services to the customers of CERC.

Approach:

In anticipation of CERC's expanding role as the national showcase of CE application for design and manufacturing, one important component in CERC's research plan is the establishment of an Advanced Prototyping Center (APC). The APC will enhance the CE product development cycle with the adoption of two missions: First, the APC will use CAD-based procedures to rapidly produce structural and electronic models of reduced size and content for the purpose of conceptual design feasibility studies. Second, the APC will provide a model testbed for conducting experiments to generate producibility data for new materials and processes. In this regard, the APC will leverage the state of CIM technology for exploring parametric links between conceptual design features and their associated metrics of producibility. The APC will also be the vehicle to bring manufacturing risk assessment to the conceptual design and evaluation process. It is different from the typical prototype manufacturing lab in that it focuses on supplying producibility information early in the conceptual design process through rapid model creation and statistically designed process experiments.

Technical Results:

Two specific goals were set for the APC's first year of development:

(i) to establish model making facilities to produce structural parts made of solid geometry metals and plastics; and

(ii) to develop a testbed for prototyping evaluation and material processing.

To adhere to budgetary and delivery constraints, four machines were selected to be best fitted with the first year APC's goals. They include:

(1) DAEWOO PUMA 6HS CNC Lathe;

(2) NUMEREX Model 2840-24 Coordinate Measurement Machine (CMM);

(3) INTERLAKEN Series 3000 Static and Dynamic Testing Machine; and

(4) PHI Model 150R30305-2LCS-P-Z(2)58 Multilayer Compression Press.

The low bay area of WVU B99 building was chosen as the APC site. WVU Facilities Planning Department was contracted by CERC to renovate the APC site and to provide the necessary power, water and compressive air line connections for the machines. All the machines are received and properly installed, and each machine is assigned to be handled by specific CERC research teams. Members of the CERC research teams will be sent to the manufacturer's training center for further advanced training on operation of the APC's machines. The training emphasis will be on learning the interface and network communication capability and programming of the machines so that, in DICE Phase III, an integrated data link between the APC and CE will be established and demonstrated. Also, the CIMFLEX Electronic Workcell will be be installed in the APC and a five-axis universal milling machine wil al;so be added to the APC in the coming quarter.

Conclusions:

As part of the DICE Phase II research effort, preliminary facilities for the APC are established. With the model making, material processing and structural testing

capabilities in the APC, some CE tools and methods developed in DICE Phase II can be evaluated and prepared for demonstration in DICE Phase III.

Recommendations:

A series of seminars will be offered to researchers in CERC other and interested industrial participants regarding the technical capabilities and functions of CERC's APC. Assistance and coordination will be provided for research teams using the APC facilities. It is recomended that development continue and that additional equipment be added to the APC so that it may become a national laboratory in concurrent engineering research.

Publications:

None.

Hardware:

Hardware purchased includes:

(1) DAEWOO PUMA 6HS CNC Lathe;

(2) NUMEREX Model 2840-24 Coordinate Measurement Machine;

(3) INTERLAKEN Series 3000 Static and Dynamic Testing Machine and

(4) PHI Model 150R30305-2LCS-P-Z(2)58 Multilayer Compression Press.

DICE Program

Task 4.2.8 **B-99 Building** **West Virginia University**

Objectives:

This task centered on the physical establishment of a comprehensive National Repository for the research, development, training and implementation of Concurrent Engineering concepts, architectures and methodology to be located at West Virginia University, on the Evansdale Campus.

Approach:

A competitive contract was awarded to a joint architect team of OMNI/WTW to develop a comprehensive schematic design with cost estimates and construction drawings to retrofit an existing building on the Campus of West Virginia University. This 35,000 sq. ft. facility will be designed to house and enhance the overall technical initiatives of the *Concurrent Engineering* Research Center (CERC).

Technical Results:

PEDCO, an engineering consulting firm, completed a third-party validation of the architect/construction documents (including estimated costs, materials and approaches) which submitted the results to the Prime Contractor.

Conclusions:

On 28 December 1989, a work stoporder was issued by the Prime Contractor halting all work efforts and planning initiatives. No further work efforts or cost expenditures were initiated, pending a third-party validation study of architect/construction documents.

Recommendations:

To continue efforts, in cooperation with the Prime, to complete the planning for the retrofit of the B-99 Facility.

Publications:

None.

Hardware:

None.

ARCHITECTURE FOR CONCURRENT ENGINEERING

Task 4.3.1 Information Content & Flow — GE Aircraft Engines

Objectives:

The goal for this task was to validate the DICE architecture in the design environment. The authors approached this task from our experiences in managing data and application programs associated with a large engineering computing environment. In this environment most design data is stored in files.

Approach: Three approaches were used to carry out the above goals:

1. Data and functional models were built to describe the design process for a particular component. These models were then examined by both the design engineers and the computer specialists in order to understand how the DICE architecture could be used to reduce the design time for the particular component.

2. A data management infrastructure was created and the common interface to engineering applications was built. These tasks resulted in the creation of a File Management System (FMS) that uses information stored in the DICE Product Process Organization model (PPO) to provide keyword based file versioning and retrieval.

   Selected design applications were "wrapped" or interfaced to the File Management System to take advantage of the keyword based file access capabilities. Design reviews and simple demonstrations were performed for the design engineers in order to assess the value of our approach.

3. The authors participated in the DICE Demonstration Committee.

Technical Results:

1. Data and function models were published which illustrate the complexity of the current engineering environment. The definition of the data and function models allowed the design engineers to develop a greater appreciation for the discipline required to manage engineering data while communicating the payoff of such a discipline.

   The scope of the logical data model was limited to engine bypass duct data. In particular, it was limited to the data utilized and produced by programs selected by the design engineers. The model provides a single logical view of the design data that is currently distributed among many separate data files. It represents the natural structure of the data independent of any application view. The model provides a blueprint from which physical databases can be built.

The function model was constructed from the viewpoint of the duct designer and centers on the conceptual design of the component. The model illustrates the complexity of the current duct design process and is a basis for understanding the savings involved in moving toward a concurrent engineering environment.

2. The File Management System was designed and a working prototype was built. The design makes no assumptions about what type of engineering activities are performed or what type of operating system is used. In particular, the approach taken is feasible under the Unix and VMS operating systems, the two major standard operating systems in Engineering.

   The FMS uses the remote query and update of the PPO Catalog to perform keyword based file access. The keywords and life cycle state are user defined and provide an alternative to the traditional path, filename, version approach. The PPO Catalog contains meta data describing file classes and file instances. The Catalog also provides the capabilities required to describe application programs and their input/output files as used in the engineering process. Distributed application interface services are provided to access the Catalog. The Application Interface or "wrapper" is language based.

   A language interface was chosen because it offers three distinct advantages over a procedural interface:

   (1) A language interface provides an easy way to specify information whose length and content may vary from one application program or user to another.

   (2) A language interface hides internal communication details from application programs and users.

   (3) A language interface supports the notion of storing file descriptive information as headers of data files so that automatic generation of Catalog entries can be obtained from the data files themselves.

   The design of the FMS was drafted, reviewed with the design engineers and then published. A working prototype was built and tested. Selected design application programs were then "wrapped" to illustrate data sharing via the FMS.

3. Data management scenarios were developed to support the DICE demonstrations. These scenarios use the File Management System to demonstrate the feasibility and value of keyword based file access methodology.

Conclusions:

1. The models revealed the following information about the design process:

    Opportunities for Data Integration: The modeling of the as-is design process identified steps of the design process that can be integrated. At present, data produced by some of the design steps are manually transcribed into the subsequent step of the analysis. These transcriptions are time consuming and disrupt the exploration of design alternatives.

    Simple Designs Require Large Amounts of Data: Large amounts of data are generated and analyzed during the design process. The conceptual design of the shell wall of a simple bypass duct involves over 300 parameters. More complex designs require tens of thousands of parameters. This data must be organized by the designer in order for the design space to be explored.

    The Design Process is Not Fixed: The modeling process also showed that, in general, designers do not follow a rigidly defined design process. This was evidenced by the many difficulties encountered by the modeling team in converging to an as-is model which was acceptable to the designers.

    Opportunities for Process Integration: At present, the design analysis involves the execution of many discrete programs. The need to examine many design alternatives requires the execution of a series of these analysis programs with varying parameters. The design engineers do not presently have a system architecture at their disposal that supports the automatic invocation of these programs and automatic plotting of results. Such a system architecture could save time currently spent writing specialized programs to plot specific groups of alternatives.

    Life Cycle Management: Central to the duct design process is the concept of data evolution. Initial design parameter values are modified by the designer as analysis is done and approvals are sought. Sometimes design data retrieval is complicated by the fact that the designer is not aware of the status of existing data files. In the concurrent engineering environment this process is complicated by the fact that many designers may be working related problems simultaneously.

2. The benefits of keyword based access of design files are:

   (1) The ability to retrieve data created by other designers without requiring personal knowledge of the work performed by others.

   (2) The ability to access engineering data rapidly.

   (3) The ability to positively identify engineering data according to its intent and applicability.

   (4) The ability to associate related design files in ways that are meaningful (configurations, tasks) to the engineering life cycle, to the engineering process and to designers.

3. The meetings held with the engineers pointed to a need for training regarding concurrent engineering in general and specifically the DICE architecture.

4. Definition of the Product Process Organization model is critical to the success of DICE. The model serves as the data integration facility and data integration is a major weakness of current design systems.

<u>Recommendations:</u>

1. Provide training on concurrent engineering concepts, the DICE architecture and specific DICE tools.

2. Expand the DICE architecture to support the configuration management of data located in files and databases.

3. Give a high priority to integration between the Product Process Organization model, the File Management System and ROSE so that a complete data integration toolkit is available through the DICE architecture.

<u>Publications:</u>

None

<u>Hardware:</u>

None

DICE PROGRAM

Task 4.3.2.1 Design Fusion Workstation **Carnegie Mellon University**

Objectives:

The long-term goal of the Design Fusion project is to develop a design system that incorporates the underlying theories, methodologies, and techniques necessary for a computer-based, integrated design environment. This system will assist the designer in creating electro-mechanical parts to ensure that they meet their function, material, cost, and quality requirements while simultaneously meeting the constraints imposed on the design throughout its lifecycle: manufacturing, planning, distribution, field service, etc. Our approach is to fuse the functional requirements of material and mechanical design with the life-cycle constraints through the active use of life-cycle knowledge throughout the design process from preliminary to detailed design. The initial domain for the Design Fusion project is turbine blade design for which functional, material, structural, aerodynamic, and manufacturing concerns must be integrated.

During Phase 2, our objective was to demonstrate a Designer's Workstation that critiques user-specified designs based on knowledge from a selected set of life-cycle perspectives and that aided in the selection of prior designs. With our prototype of the Designer's Workstation in place, we have explored the issues of coordination, of constraints, and of representations that support mechanical designers.

Approach:

The architecture for the Design Fusion system is based on a blackboard model. Each design, composed of features connected by constraints, is represented in the *design blackboard.* Perspectives are represented as knowledge sources that can view the designs in parallel. In Design Fusion, each design perspective can both criticize design decisions and suggest design changes.

Knowledge-based systems and expert-systems technologies provide software architectures that can assist the mechanical designer. The role of the architecture is to integrate the algorithmic and heuristic processes used during design. Through integration, methods employed by designers can communicate via a common representation of the design.

The design model is integrated around a shared, domain-neutral representation of the design from which each perspective can extract and reason about features of the design. Constraints are the language by which perspectives communicate with one another and with the designer.

The perspectives are coordinated through a blackboard architecture that uses a heterarchical control structure. Our system is based on the concept of a shared representation. The shared representation of the design is maintained on the blackboard, and all comments, constraints, and design changes are made in terms of it. Our architecture does not preclude a perspective from creating its own representation, but communication is always through the shared representation.

In the context of Design Fusion, the individual design perspectives (e.g. structures, aerodynamics, manufacturing) communicate their considerations by asserting constraints among feature parameters. Although in principle an acceptable design may be obtained by solving the constraints, practically the number and complexity of constraints arising from these considerations make it difficult to identify satisfactory designs. Furthermore, simply solving the constraints provides no information as to which constraints limit the cost or quality of the design.

We address these difficulties by reasoning about the structure of interacting constraints so that we may simplify the process of constraint satisfaction and so that we determine which of the constraints are relevant, redundant, or inconsistent. Characterization of constraints further simplifies the problem of identifying an acceptable design and provides focused feedback to the design perspective.

During the design process, large quantities of information about a design are used and generated. We have made the decision to include in the shared representation only those attributes which are of interest to more than one perspective. Using perspectives enables us to partition the design knowledge into manageable chunks, while allowing us the flexibility to add new information to the representation. For example, the manufacturing perspective may have a constraint on the maximum length of a cast turbine blade. As long as this constraint is not violated, it remains within the perspective; however, if it is violated, the manufacturing perspective would post the constraint on the blackboard.

Technical Results:

We have implemented a blackboard architecture for the task of maintaining a common representation of the design and coordinating the various perspectives of the design process. The blackboard consist of three panels:

- a geometry panel on which a geometric representation of the design is maintained
- a constraint panel on which the design constraints are maintained
- a database panel that maintains the qualitative design features and quantitative design parameters derivable from the geometry.

A protocol has been established for the blackboard that provides a means by which the

perspectives can both communicate with each other and affect the design.

The multi-perspective, constraint-based paradigm of design results in a large set of complex constraints; however many of the constraints are redundant (e.g. a size constraint for transportation is likely to be redundant in light of a size constraint imposed to facilitate manual assembly). Redundant constraints do not influence the design and can be neglected. Other constraints direct the design: Requiring a tank with specified volume and minimal surface area caused the designer to select a spherical tank.

One of the two thrusts of our constraint related work deals with the characterization of constraints in this way. Specifically we have:

- Developed and implemented an interval based approach to monotonicity analysis to facilitate the identification of constraints that are critical and therefore direct the design
- Developed and implemented interval based techniques to identify active constraints and those which are unconditionally inactive.
- Implemented strategies based on interval dominance to identify the dominant constraints in a conditionally critical set, thereby simplifying the set of relevant constraints and the corresponding constraint satisfaction problem.
- Implemented interval dominance techniques to prevent the combinatorial explosion associated with an active set strategy.
- Identified a negotiation strategy for conversation across perspectives and proved activity and criticality of newly introduced constraints.

Although most constraints interact with almost all other constraints indirectly, their direct interactions are often sparse. This characteristic of large sets of design constraints can often be exploited to identify a sequence of design decisions which minimize the degree of iteration required. This has been the focus of the second thrust of our work on constraints. We have:

- Developed two heuristic algorithms for selecting variables best suited to resolving simultaneous constraints. We use a *strong component* based measure of coupling to select variables.
- Completed implementation of graph-theoretic algorithms for planning the solution of a combined set of implicit and explicit constraints. The planning algorithm can now handle conventional algebraic constraints as well as black-box constraints (e.g. an FEM package and non-invertible algebraic relations.
- Devised an analytical method based on the boolean adjacency matrix of the constraint set to determine whether the set is serially decomposable or partitionable. The structure of the set is reflected in the properties of a determinant expansion of the adjacency matrix. A complete identification of the optimal partitions can be done by examination of some properties of the determinant expansion.

Our research in feature-based representations of designs has been motivated by the realization

that geometric models represent the design in greater detail than can be utilized by designers, process planners, assembly planners, or by the rule-based systems that emulate these activities. Experts often abstract geometry into features like ribs, parting planes, and chamfers; however, the same product design looks quite different when viewed by different experts. Each perspective emphasizes particular aspects of the design and suppresses certain details in order to evaluate and synthesize. In addition, as the design evolves, so does the view from each of the perspectives; that is, what is emphasized and what is suppressed changes depending on the current state of the design.

Because each perspective views the design differently, each perspective defines its own set of features. And, because the features are defined in terms of the shared representation, the perspectives can communicate by referring their features to the shared representation.

We have worked on integrating the representation of the geometry of a design with the representation of the symbolic attributes of a design, such as taxonomies and relationships. Our research on exploiting the graph-based representation of geometry in Noodles for feature description and feature extraction is promising. This approach has enabled an elegant and consistent representation of apparently disparate attributes of a design.

To date, our research has focused on defining and recognizing shape features, that is, features that are derivable from the geometry and topology of the design. Our approach to feature extraction is to describe features using a graph grammar. Because the designed object is an element in the language generated by this grammar, features can be recognized by parsing the feature graph against the graph of the object. We provide a representational link between the low-level geometric representation and the high-level design abstractions by formalizing a language to express classes of high-level objects in terms of low-level ones. Given this language, we can extract high-level elements from low-level geometric representations.

Conclusions:

We have implemented the second version of the design system that embodies the research presented above. This system, known as Design Fusion, has enabled us to test and refine our ideas on concurrent design. In the process of implementing the Design Fusion system, we have

- created a method for defining and recognizing non-manifold features and have begun to implement an efficient algorithm for recognizing features in an evolving design
- created an architecture that integrates partial solutions to portions of the design problem based on a common representation
- created new algorithms for reasoning about constraints using interval methods and regional partitioning.

The blackboard architecture developed for Design Fusion has proven effective in maintaining a shared representation of the design and acting as a communication medium for the various design perspectives. With this architecture designers are freed to concentrate on the design, not the process.

The constraint reasoning and solution planning methods appear to be quite powerful in dealing with large sets of complex constraints. When applied to a network of turbine blade aerodynamic and structural constraints the interval monotonicity methods identified critical constraints which directed design parameter selection. The constraint dominance methods determined that a structural constraint dominated an aerodynamic constraint. The solution planning approach applied to the resulting set of constraints determined that the design problem could be solved without iteration and was able to identify which of the constraints if relaxed would improve the the quality of the design.

The Design Fusion system supports concurrent design by enabling the simultaneous consideration of life-cycle constraints. It uses a shared representation of the design which can be parsed using perspective-specific features. It uses constraints as a language by which perspectives communicate with one another and with the designer. The perspectives are coordinated through a blackboard architecture that uses a heterarchical control structure.

Recommendations:

Although the methods described above have proven to be powerful in one context, we expect that their utility will depend on the characteristics of the design domain. We propose to extend the research in the areas of architecture, constraints, features, and geometry; however, one of our primary goals is to show the flexibility and generality of the Design Fusion system. To this end, we propose to implement the system in a new mechanical design domain.

We are currently working on augmenting the blackboard protocol to facilitate the capture of design history and version control. They provide a means by which the system can:

- revert to a prior design
- perform dependency-directed backtracking and
- build an explanation facility that can demonstrate how the propagation of constraints lead to the assignment of various feature attributes.

We believe that it is possible to characterize constraint networks a priori so as to select methods that will have the greatest utility. We also believe that less conservative measures of constraint dominance can be identified that will further enhance our ability to reason about critical design constraints and therefore provide a basis upon which to base strategies for negotiated constraint relaxation.

We also are continuing to develop an integrated representation of the design to be shared by all the perspectives and to enhance the non-manifold representation of geometry used in the Noodles model with respect to form-based design and manufacturing features. We will continue our design and implementation of the representation of:

- geometry and topology based upon the Noodles representation
- features for each of the perspectives
- design record information

Publications:

"Life-Cycle Features for Computer-Assisted Design," Scott A. Safier and Susan Finger, to be presented at the International Conference on Engineering Design, ICED, Dubrovnik, August, 1990.

"Parsing Features in Solid Geometric Models," Scott A. Safier and Susan Finger, To be presented at the European Conference on Artificial Intelligence-90.

"Representing and Recognizing Features in Mechanical Designs," Susan Finger and Scott A. Safier, To be presented at the Second International Conference on Design Theory and Methodology, Chicago, September, 1990.

"Concurrent Design," Susan Finger, Mark S. Fox, and Friedrich B. Prinz and James Rinderle, invited for a special issue of *Applied Artificial Intelligence*.

"Interval Approaches for Concurrent Evaluation of Design Constraints," D. Navinchandra and J.R. Rinderle, *Concurrent Product and Process Design*, ASME, DE-Vol. 21, PED-Vol. 36, December 1989.

The Role of Architecture in Computer-Assisted Design Systems, Scott A. Safier and Mark S. Fox, January 30, 1990

Hardware:

No software or hardware was purchased. We have created the software for the designer's workstation using the blackboard architecture that integrates the Noodles geometric model, the feature extraction algorithms, the direct manipulation interface, the experimental version of constraint-reasoning software, and the software for the aerodynamic, manufacturing, structures, and designers perspectives.

DICE Program

Task 4.3.2.5 I-BUS Architecture Control Layer West Virginia University

Objectives:

The DICE Communication Channel (DCC) is the underlying layer of communication software enabling and facilitating the cooperation and coordination involved in Concurrent Engineering. It consists of the supported communication protocol (TCP-IP) and utilities like *telnet* and *ftp* and additional monitoring facilities to evaluate the performance of the DICE services in operation. Utilities also exist to enable synchronous exchange of graphics and text among product developers.

The Concurrency Manager, embedded in each node as a module called the Local Concurrency Manager or LCM, eliminates the need for users or applications to manage remote process communication themselves. Instead, the LCM's, taken together, implement a complete virtual network of connected workstations, nodes, servers and resources whose capabilities to execute applications are known collectively to the LCM's and are available transparently to all users and applications. Implementation of a high level of transparency and the ability to execute a suite of applications in some pre-defined order on multiple workstations are goals of the CM.

The BLACKBOARD COOPERATION scheme enables DICE users to share all or part of their evolving designs with local or remote development team collaborators. This facility encourages designers who have completed their sub-tasks to publish relevant results on a blackboard. All affected designers are notified on a "need-to-know" basis, thus providing an easy mechanism for catching downstream conflicts early.

Approach:

The DICE approach to Concurrent Engineering Design visualizes several design groups working with individual CAD Tools on workstations connected in a network. In such a system, there is a large contribution to the efficiency of the concurrent design process from the computer assistance present for the three C's: Communication, Cooperation and Coordination.

In the system described, there is a layer of software running on every networked computer to enable the engineers to take advantage of the presence of cooperating

experts and their tools on the network. The chief functions performed by this software are summarized below.

Communication

New utilities to enhance cooperative communication for problem-solving are needed. This software, called *Cooperate,* connects a group of engineers and organizes a virtual "meeting" on the network. Another utility, called View, enables the communication of graphics between networked designers to buttress the exchange of design information during the meeting, and this has been merged with *Cooperate*. A new pseudo X-server is visualized to make it possible for several designers to work in round robin fashion on the same application screen.

Cooperation

Any application or designer can invoke the CM in his workstation to communicate with any other application or designer in the network. A complete protocol labeled as the Communication Services (CS) is defined.

The CM also implements the ability to have a local menu and window interface to another designer's application without need of login authority or special expertise. This is called the Application Management Services (AMS).

An important element, called the Network Services (NS), is used as a distributed slowly-changing data base for holding the network configuration, the application services of the network, cataloged job definitions and so forth.

A fourth and equally important capability is that of running a job consisting of numerous programs in different computers in a defined precedence relationship. Designers who can invoke such an execution capability, leaving the CM to manage the details, have a powerful tool to perform network scheduling of tasks. This is called the Task Management Services (TMS).

Coordination

The third component of the systems software is aimed at making the activities of the designers and the Project Leader (PL) visible to each other. A modified Blackboard architecture is employed to implement the coordination between designers. The PL can assign tasks and access the engineering data base to bring down parts of it to the globally visible blackboard workspace. The blackboard maintains the current state of the design and designers are expected to assert any proposed changes to the design on the blackboard. Conflicts are detected by the blackboard. The main purpose is to allow designers and the PL to exploit the global visibility of the blackboard as a

coordination mechanism. The ability to display the product structure under evolution in a graphic form using trees and user interaction at the nodes of the tree is a new enhancement.

The net effect of implementing the three C's is to increase the level of concurrent activity in the network.

Technical Results:

Under Communication, the second versions of the *Cooperate* and *View* utilities were implemented and demonstrated at the December 15, 1989 demonstration.
The Communication Monitor was also implemented in a base level version and the capabilities demonstrated in a stand-alone mode on Unix workstations of the Sun variety.
The Concurrency Manger was used in both the demos on December 15 for its CS services among three varieties of Unix workstations. The NS, AMS and limited TMS services will be displayed in the September 1990 review for DARPA.
The DICE Blackboard for Design Evolution played a small role in the December 15 demonstration for distribution of tasks to perspectives. It will be capable of a much greater role in future demos because of new developments under way, such as the Design Assessment Tool, the Graphic Product Display mechanism and forms for Tasks and Assertions.

Conclusions:

The approach was shown to be useful. Further work will deepen the scope of the modules and port it to other UNIX platforms and the VMS platform.
The possibility of integrating with other modules in many ways was also proven. However, the actual smooth interchange of data is yet to be achieved and will be the focus of the September 1990 demo.

Recommendations:

The tasks undertaken must continue to be supported. In their full versions, by December 1990, they will form the core of a useful set of DICE services. Porting to numerous platforms, including VMS, will increase the use of these modules.

Some further tasks emanating from our current developments were proposed in Phase III for funding -- for example, support for PC platforms. They are quite important to the ultimate transitioning of DICE to industry.

Publications:

A paper from the Architecture group was submitted to the MIT conference on Cooperative Product Development in November 1989:

F. Londono, F., K.J. Cleetus, and Y.V. Reddy."A Blackboard Scheme for Cooperative Problem-Solving by Human Experts."

Three papers from the Architecture group were submitted to the Second National Symposium on Concurrent Engineering in Feb 1990:

Londono, F., K.J. Cleetus, and Y.V. Reddy "A Blackboard Problem-Solving Model to Support Product Development."

R. Raman, R., K.J. Cleetus, and Y.V. Reddy "The Local Concurrency Manger in Distributed Computing."

Coleman, James M., and William H. Dodrill "DICE Network Monitoring "

A technical report from the Architecture group was submitted to the CERC Library repository:

Cleetus, K. J. "An Introduction to the Structures and Electronics Demonstrations."

Two problem reports for Master's theses were submitted to the Dept of Statistics and Computer Science:

Woo, Tony Chi Hung. "DICE Callable Interface Procedures Using X and TAE Plus."

Tumalapalli, Rao. "The Application Management System of the Concurrency Manager."

Hardware:

Software purchased includes TAE Plus.

Software developed includes:

- Cooperate View;
- DCC Communications Monitor;
- Blackboard for Design Evaluation (DBB); and
- Concurrency Manger (CM).

No developed software is yet in a form releasable to alpha sites.

DICE PROGRAM

Task 4.3.2.6 DICE to Relational Database Link

Rensselaer Polytechnic Institute

Objectives

The typical design system contains many databases several of which may be relational databases. The objectives of this task were to try to categorize the roles of these relational database systems, and to construct a link between those systems and the intelligent file cache.

Approach

The role of legacy database systems in concurrent engineering was examined by performing an informal survey of existing design systems and their databases. Relational database systems were found to exist in configuration control systems, data dictionary systems, logistics systems, manufacturing information systems and data repository systems. We examined these systems with the aim of formalizing their role with respect to the intelligent file cache, the PPO and concurrent engineering

A prototype intelligent file cache to relational database link was implemented by taking data from a file cache application and examining how that data might be put into a relational database. The application examined was a circuit design system implemented for the US Air Force. We looked at the kinds of information that could be extracted from the database of that system and put into a relational database with the aim of using that relational database as an index into the circuit design system's database.

Technical Results

The result of our examination of legacy database systems and their role with respect to the PPO and the intelligent file cache is shown in Figure 2. As this figure shows, the PPO of a concurrent engineering system can be divided into four basic components: a data dictionary, a configuration control system, a change management system (the intelligent file cache), and a schema generation tool (PDES EXPRESS).

Today, the data dictionary is stored in a relational database system. In the future it may be stored in either a relational database or an object oriented database system. The FMS and the IRDS standards are examples of systems of this type, but any relational database that stores catalogs or transaction processing information is included in this category. Support for transaction processing is an important characteristic of this type of system. It means that the client applications of a data dictionary system always have the same value for the objects in a database. This is important in manufacturing, for example, if two processes attempt to credit or debit a "Quantity on Hand Field" at the same time. To make sure that both processes agree on a single value for the field, the dictionary system will halt one of the processes and make it wait until the other process has finished its credit or debit.

The configuration control system is used to control a database of designs. Such systems are typified by the WISE system of Westinghouse, the Sherpa system and in software engineering by SCCS and RCS. These systems maintain catalogs of old design versions. They can perform quality control by only allowing a design to be promoted to a higher status if it passes a prescribed series of tests, and they can select which version of a design should be delivered to a user according to his status within a hierarchy. Many configuration control systems store catalog data describing their configurations in data dictionary systems. In concurrent engineering, the role of a configuration management system is to enforce the policy of an organization with respect to the selection and promotion of design versions.

The change management system is used to represent designs, to identify conflicts between design versions and to propagate changes between design versions. The intelligent file cache is an example of a change management system. A change management system describes design versions in a way that allows those versions to be read into multiple programming languages, it identifies conflicts between design versions as delta files, and it propagates changes between design versions using delta files. The change management system allows the PPO to manage changes using publish and read protocols. These protocols are more appropriate for managing design changes than the synchronization protocol enforced by transaction processing systems because they allow one user to publish a change to a design, and another user to read that change. Therefore, both users get a chance to accept or reject the changes.

The schema generation tools of the PPO are used to describe the format and organization of a database. The PPO describes a relational database to the data dictionary system as a set of relations in SQL. It describes a design database to the change management system as a set of designs using PDES EXPRESS, and it describes configurations to the configuration control system as a set of catalogs using the language of that system†.

An application program accesses objects by value, design name, or by Object ID. It accesses objects by value by sending a query to the data dictionary system, it accesses them by design name by sending a query to the configuration control system, and it accesses them by Object ID by sending a query to the change management system. Each system is responsible for delivering an object to the application program as quickly as possible within the constraints defined by the types of services it provides its users. For example, the data dictionary system must check secondary storage to make sure that no one else is changing the object, but the change management system will have put a copy of the object into memory because any changes made to the object by other users will be published as a delta file that can be read by the application at its discretion.

In the architecture, the change management and configuration control systems use the data dictionary system to find designs when the name of that design is not known. Relational databases have powerful query languages such as SQL that can be used to find any record using any combination of values. Although it is possible to apply such languages to product data described using PDES EXPRESS, experiments show that in most circumstances this is not useful for real data because that data is so detailed and difficult to understand outside of the context of an application program. Instead, it is better to extract summary information from design data and put it into the relational database so that it can be used to find the designs that contain that information. As part of this task we constructed such an index by writing a program to extract high level, easy to query data from a circuit design database. The results of our experiments showed the general form that a relational database index might take, and demonstrated that a relational index would have acceptable performance with respect to finding designs, but that the range and form of the index may have to be limited to make sure that the

†At present, there is no standard for defining catalogs to configuration control systems.

relational database can be updated in real time when a user publishes changes to a design.

Conclusion

A PPO for concurrent engineering can be divided into four systems as shown in Figure 2.

1. A data dictionary system that provides transaction processing style concurrency control for record oriented data.
2. A configuration control system that stores and retrieves versions of designs according to locally defined policy.
3. A change management system that provides "publish and read" style concurrency control for design oriented data.
4. A schema generator that supplies PDES EXPRESS and SQL descriptions of concurrent engineering data to all three systems.

Recommendations

The intelligent file cache and relational database systems should be enhanced so that an application program can store data in either system as appropriate. Standards committees call this issue inter-operability. Further research should be performed within the DICE program to decide the requirements of such inter-operability from the perspective of concurrent engineering.

The DICE program should continue work on making it easier to migrate data from the intelligent file cache to the data dictionary and vice versa. Concurrent engineering enterprises will use change management systems to manage changes to designs, and data dictionary systems to synchronize multiple manufacturing processes. As a product moves from design to manufacture, data will need to migrate from the change management system (design database) to the data dictionary system (manufacturing database).

Publications

[1] "Using a Relational Database as an Index to an Object Oriented Database in Design Applications", Hardwick M and Samaras G, Second International Conference on Data and Knowledge

Systems for Manufacturing and Engineering, at NIST in Gaithersburg, October 1989, available from the IEEE Computer Society Press.

Hardware

The work performed for this task used the same hardware as that of Task 4.3.8.1

DICE PROGRAM

Task 4.3.2.7 Intelligent File Cache (Part A) — Rensselaer Polytechnic Institute

Objectives

The objective of this task is to implement a data management system for concurrent engineering that allows DICE users to access designs from a variety of different programming languages and to operate on those designs concurrently with other users. The system should not slow the performance of an engineering application below acceptable levels. It should allow users to make changes to a design concurrently, and it should contain tools to control and propagate the effect of changes.

Approach

The file cache is implemented as an extension of file system technology. It allows clusters of objects representing versions of designs to be down-loaded into an application program. Clustering objects into designs allows an application program to translate the internal references within a design into main memory pointers that can be traversed more efficiently than indirect pointers. Clustering objects into designs also makes configuration control easier because it is clear which objects must be replaced by a new version when a design is replaced by a new version.

A design database in the intelligent file cache can be accessed as collections of objects using class libraries implemented in three languages: C++, Objective-C and LISP/CLOS. In each language an engineering design is presented to an application program as a set of objects that can be traversed in accordance with the requirements of that application program. These objects can reference each other, objects in other designs, and objects in other database systems. The class libraries have been written so that they can be added to C++, Objective-C or CLOS programs with the minimum of effort.

The intelligent file cache can compute the difference between two design versions as a delta file. Each record in a delta file describes an edit to an object in a design. Taken together, these edits describe how one version of a design can be converted into another version. The file cache allows a delta file to be generated as an application edits a design, or by using a tool to compare two different versions of a design. In the first case the delta file may contain a large number of redundant edits. In

the second case the delta file will be minimal. In either case the delta file can be generated without any additional effort from the application programmer.

Delta files allow users to develop designs concurrently because they make their changes to a design and then send those changes to other users who are also working on the design. This can be done as quickly as possible so that the users appear to be editing one design version in parallel, or more slowly so that each user can be sure that his edits are correct before they are passed on to the next user.

The full list of roles for delta files is still being researched. Other potential roles that we are aware of include:

1. Making an application reliable in the event of a system failure. As an application program edits a design, a delta can be generated and stored in a file so that the latest value of the design can be recreated if the application fails for any reason.

2. Audit trails. A delta file records all of the changes that a user has made to a design including the redundant changes. Therefore, it can be used as a very detailed audit trail of all the changes that were tried including those that were rejected as being unsuitable.

3. Data compression. If an organization needs to store two large versions of a design, then the volume of data that must be stored can be reduced by storing one version as a delta to another version.

4. Conflict resolution. If the delta computed as the difference between two design versions is non-empty, then those two versions have conflicting values in some of their features. A user can look at a delta file and use it to decide which version of a design should be changed for each conflict.

5. Views. If a delta file transmitted between two applications is modified as part of the transmission, then those two applications have different views of the same data. For example, a circuit schematic design system and a circuit layout design system may be editing different but similar designs for the same user.

6. Change notification. A delta describes a change to an object in a design. If the changed object is a critical parameter in the design, then the delta describing its new value can be transmitted to

other users as appropriate.

7. Instance inheritance. A design may use another design as a sub-design in many different places. Each instance may need to customize the sub-design in different ways. For example, different instances may need to give different values to text labels describing the inputs and outputs of a circuit. These customizations can be described as delta files that an application program applies to a sub-design immediately before it is instantiated into a parent design.

Technical Results

The Intelligent File Cache has been implemented so that every design is stored as a structured file whose schema is defined by the PPO. The objects in a design can be accessed transparently using their Object ID's from applications written in three programming languages: C++, LISP/CLOS and Objective-C. Delta files can be generated transparently as an application edits a design in two of these languages: C++ and Objective-C†. In each language the interface to the database is presented to the user as a class library that can be sub-classed using the standard techniques of object oriented programming.

Benchmark results have been generated for the system using data taken from a circuit board design system. The circuit board that was tested was considered to be large three years ago. In the file cache it is stored as a file containing 157,950 objects in 5,373,175 bytes of storage. An algorithm was written to select the components in the circuit board that are visible in a display window. When the algorithm was completed, it took 0.03 seconds to select those components for a typical window, after the board had been loaded into memory. To achieve this result, the algorithm had to be coded so that it only tested objects that could be visible in a window. The data and algorithm used in the test were given to us by a computer vendor. The tests were performed on a DEC Station 3100 with 24 megabytes of memory. The result shows that the file cache does not slow the performance of an application as it traverses the objects in a design.

One criticism of the file cache is that it requires all of the data in a design to be loaded if any object in that design is needed in an application. Loading all of the data in a design takes time and

†The third language will be enhanced to read and write delta files as part of Phase III.

there is a possibility that a design could become too big to fit into virtual memory. The file cache relies on its users dividing their designs into sub-designs before they become too big for virtual memory. The time to instantiate all of the objects is a problem that makes the file cache unsuited to transaction processing applications. In design, many applications have algorithms that must visit most of the objects in a design to compute their results. For these algorithms loading all of the data in a design immediately is an advantage. For example, in many systems the first operation that must be performed is a display operation that requires a significant proportion of the objects in a design to be visited.

On a DEC Station 3100 using the data of the 5 Megabyte circuit board design, we measured the C++ programming interface of the cache as being able to read all 157,950 objects in 43.29 seconds and write those objects in 29.66 seconds. This is equivalent to a read rate of approximately 3,000 objects per second and a write rate of approximately 5,000 objects per second. The numbers shown were obtained by making calls to the UNIX time function and by adding the system and user time components of that function.

The read and write times shown here are limited by the CPU speed of the workstation, by the speed of the network that connects the workstation to its file server, and by the burst (data transfer) rate of the DASD device that stored the file. The speed of the network was not a problem in the measurements made because it was able to deliver packets to the workstation faster than the disk system could read them or the CPU could process them. CPU and DASD burst speeds are improving quite dramatically with each generation of workstation and disk device. The DASD seek times that are critical for indexed database systems, such as relational database systems and object oriented database systems, are not improving at the same rate.

Scripting has been demonstrated in the file cache using a tool to import an IGES file and show how two users can edit that file concurrently using a simple picture drawing system. This demonstration shows how two users can edit a drawing at the same time on different workstations. It also shows how the changes made by those users can be merged into a single drawing, and how a lead engineer can identify and resolve any conflicts between the edits made by those users. The scripting part of the system is still being researched and developed. Therefore, it would be premature to release performance

results for this part of the system, because any figures given would be subject to significant improvement, and because many of the roles that scripts can play have not yet been fully developed.

The architecture of the file cache as it exists today is shown in Figure 1. Note that all of the tools shown in this figure exist except for the applications shown in the top layer of each workspace. These applications are given as illustrations of the types of applications that might use the file cache. In the figure, the cache is described using its Rensselaer acronym of ROSE, and the different language versions are shown as ROSE++ (C++), ROSE-IC (Objective-C) and ROSE-CL (Common LISP)

Conclusions

An intelligent file cache can be implemented for concurrent engineering that:

1. Does not significantly reduce the execution speed of an engineering application.
2. Supports systems and applications written in multiple programming languages.
3. Allows users to edit the same design concurrently and share the results of their edits using delta files.
4. Identifies the conflicts between design versions as a delta file so that an engineer or application can resolve those conflicts.
5. Propagates changes between design versions using delta files.

Recommendations

The PDES EXPRESS language is an emerging standard for describing product data. In the context of the intelligent file cache it represents another object oriented programming language that should be supported by the cache. In this case, the language is one that can be used to define but not manipulate data. Clearly, one item for further research must be to investigate how the intelligent file cache can be extended so that it accepts EXPRESS definitions of database models and allows databases described by those models to be manipulated in C++, Objective-C and CLOS.

The work done so far on the intelligent file cache demonstrates that describing and propagating changes between design versions as delta files has great value in concurrent engineering. Research should continue to investigate further roles for delta files and produce quantitative results that compare

different ways of implementing those files. Where appropriate, the work should compare the results against similar results that can be obtained using other tools. For example, the SCCS source code control system of UNIX can generate delta files for text applications.

The file cache as it has has been implemented is a first prototype. More work is needed to make the concepts in this prototype easier to understand, to add missing functionality, and to remove unnecessary functionality.

Publications

[1] "ROSE: A Database System for Concurrent Engineering Applications", M. Hardwick, D. Spooner, E. Hvannberg, B. Downie, J. A. Faulstich, D. Loffredo, A. Mehta and D. Sanderson, Second Annual Conference on Concurrent Engineering, West Virginia University, February 1990.

[2] "The Evolution of ROSE: An Engineering Object-Oriented Database System", D. Spooner, M. Hardwick, E. Hvannberg, B. Downie, D. Loffredo, A. Mehta, J. A. Faulstich, D. Sanderson, R. Harris, G. Abou-Ezzi, J. Gong, J. Young, M. Rovira, and G. Samaras, Proc. of the Second Rensselaer International Conference on CIM, May 1990, available from the IEEE Computer Society Press.

Hardware

A network of DEC Station 3100 machines was purchased to develop the software in the intelligent file cache. This network consists of a file server and 4 workstations each with 24 Megabytes of memory. The file server has DASD devices able to hold 1 giga-byte of data.

Task 4.3.2.7 File/Database Translator (Part B) Rensselaer Polytechnic Institute

Objectives

The Initial Graphics Exchange Standard (IGES) and the Product Exchange using Step (PDES) standard provide ways to import and export data between engineering systems. However, the data that has to be imported or exported may not have the same form as that used internally by the engineering system. Therefore, these systems will need to be able to translate their data from their internal format to an external format and vice versa. The objective of this Task is to make this task as easy as possible.

Approach

Data dictionary systems can use SQL to translate their data. In the PPO, design data is stored in the Configuration Control system but manipulated by the Change Management System. This system receives PDES EXPRESS descriptions of a database from the schema generator and makes those descriptions available to an application program as objects defined by classes in C++, Objective-C or LISP-CLOS. The data of PDES EXPRESS and Object Oriented languages is detailed and difficult to translate using SQL. Application programs want to edit such data by sending messages to objects and not by calling SQL. Therefore, we decided to implement a translator that relied on sending editing messages to objects.

A message oriented approach has the advantage of allowing edits to be applied to individual objects or to collections of similar objects that have been found using some algorithm of the application program. In a message oriented approach the translator can be presented to the user as a series of operations that will add, subtract or move data fields between objects. Hence, users can be given fine control of the operation of a translation program. If a user needs to perform a mundane translation, and does not want to be bothered with the details of application programming, then this can be handled by generating a program to perform the translation.

Technical Results

A data translator has been implemented as a set of classes in Objective-C that will add, delete and move data fields between classes of objects. The data translator is written in Objective-C but all objects in the intelligent file cache can be accessed from all three languages supported by the cache, so the translator can be used to translate objects generated by C++ and CLOS programs.

The data translator converts designs between two versions of a schema. As shown in Figure 2, the two schemas can be generated using the PPO schema generator. Hence, the data translator can be used to convert designs between versions of PPO schemas. If the differences between the schemas are trivial then the conversion program can be generated automatically, and if they are complex then the program must be written by a programmer using the class library of the translator. For example, if an application program contains a class of objects that model lines as two end points, and another application program requires its lines to be modeled as four coordinates, then a data translation program can be written using the data translator to perform this operation. For simple edits such as this, a data translation program can be generated automatically from two schema descriptions. For more complex edits where only some lines have to change according to some selection criteria, or where fields have to move between loosely connected objects, a user program must be written to call the translation classes.

The data translator has been tested using data imported into the intelligent file cache from CAD systems using the IGES data exchange standard. The data was translated into a different, more easy to use form and exported to another application of the file cache.

Conclusions

A data translator that converts engineering designs between different data formats can be written as a set of classes that manipulate objects. For simple changes a translation program can be generated automatically. For complex changes where the edits must depend on the semantics of an application, it can be done using a class library that implements a series of editing functions. If the different data formats are described using PDES EXPRESS schemas, then the data translator can be used to translate designs between those schemas.

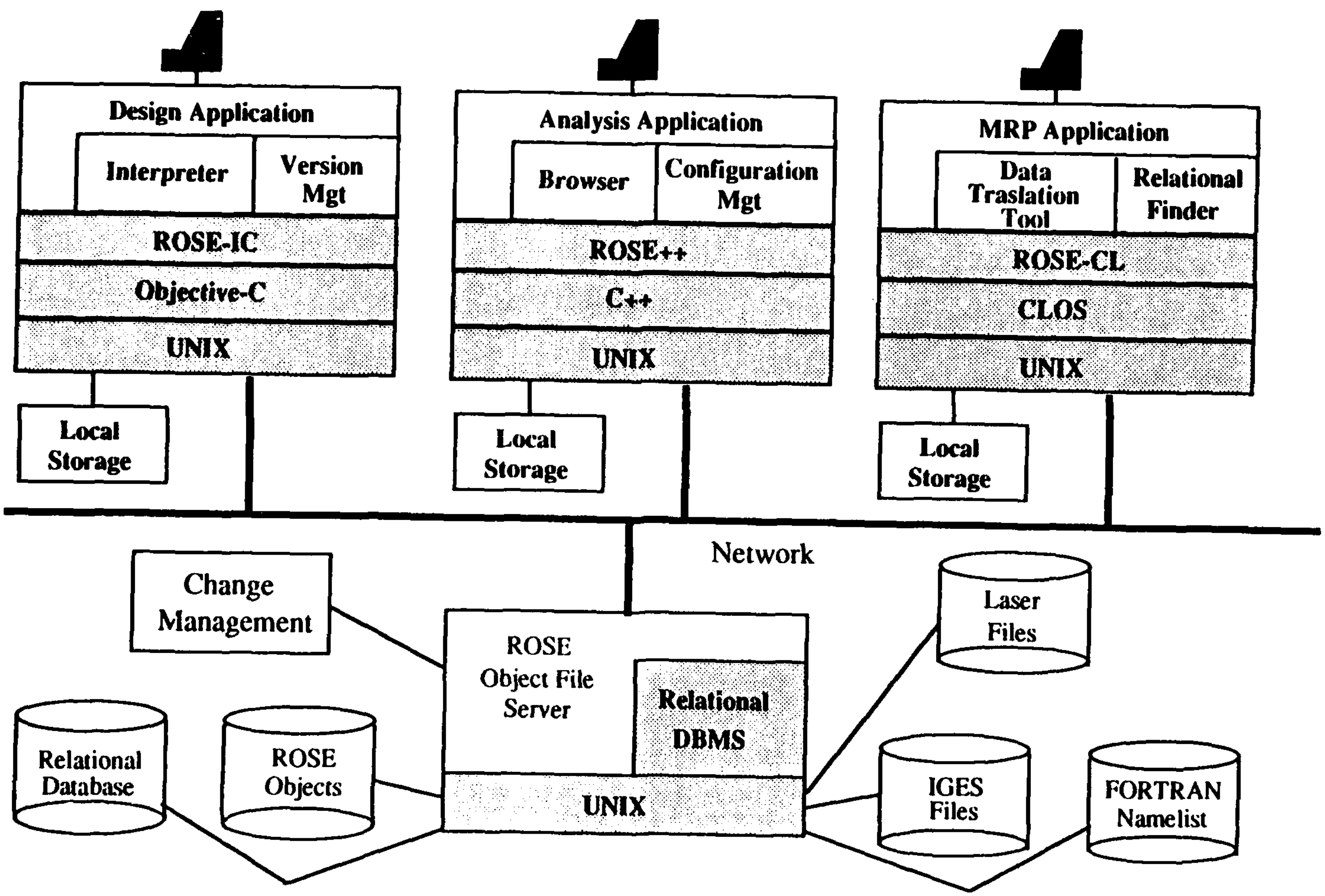

Design Application
Interpreter
Version Mgt
ROSE-IC
Objective-C
UNIX
Local Storage
Analysis Application
Browser
Configuration Mgt
ROSE++
C++
UNIX
Local Storage
MRP Application
Data Traslation Tool
Relational Finder
ROSE-CL
CLOS
UNIX
Local Storage
Network
Change Management
ROSE Object File Server
Relational DBMS
UNIX
Relational Database
ROSE Objects
Laser Files
IGES Files
FORTRAN Namelist

Figure 1

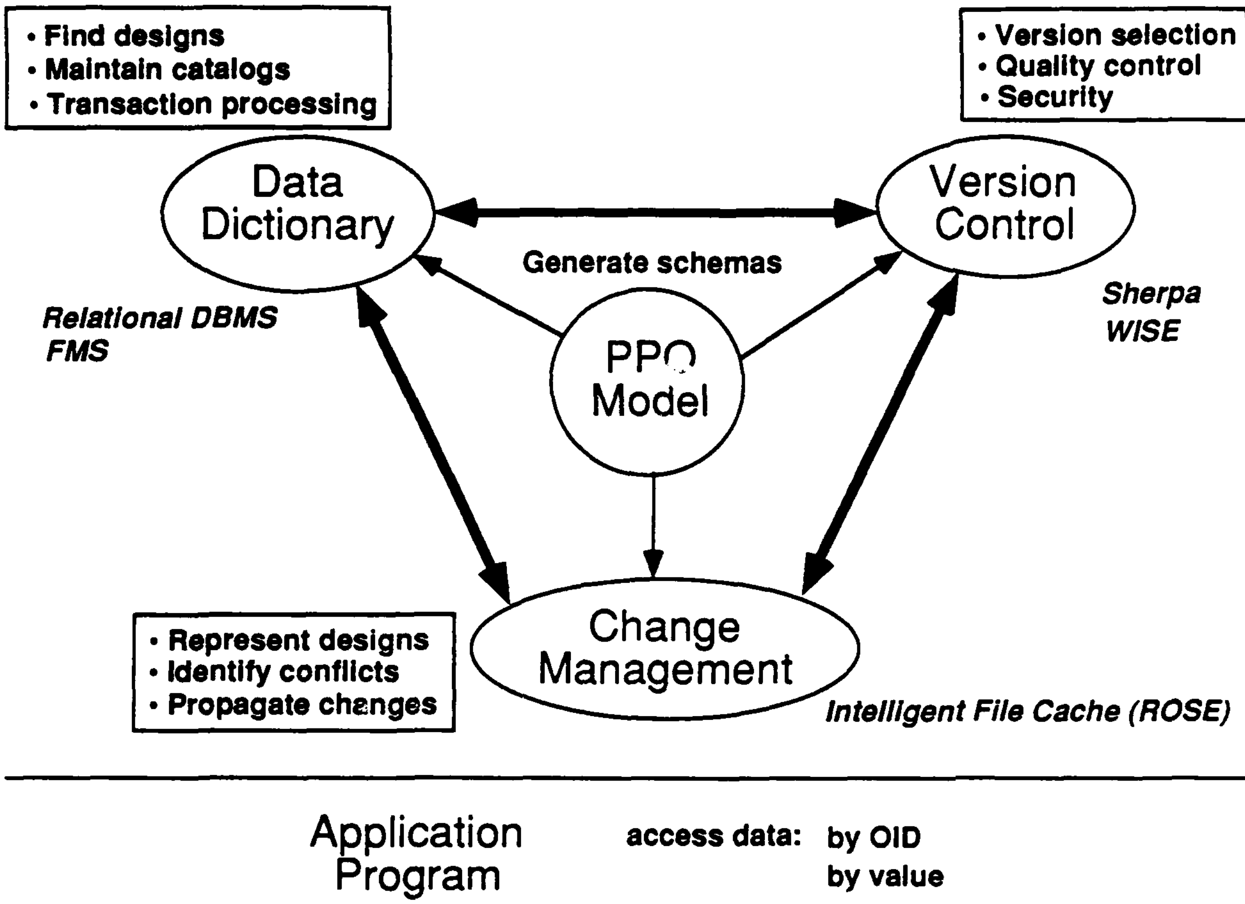

• Find designs
• Maintain catalogs
• Transaction processing
• Version selection
• Quality control
• Security
Data Dictionary
Version Control
Generate schemas
Relational DBMS
FMS
Sherpa
WISE
PPO Model
• Represent designs
• Identify conflicts
• Propagate changes
Change Management
Intelligent File Cache (ROSE)
Application Program
access data: by OID
by value

Figure 2

Recommendations

The data translator should be extended so that it can convert data between versions of schemas described using PDES EXPRESS. More research is needed on this item, even though we have demonstrated that it can be done in principle, because PDES EXPRESS can describe more of the semantics of a database than IGES, and a user may not want to loose those semantics in a translation. For example, PDES EXPRESS can constrain the sizes of lists and in a translation it may be desirable to convert a fixed sized list into a record with specific fields allocated to each list item.

Publications

[1] "A Data Translator for Engineering Systems", D. Spooner, D. Sanderson and C. Charalambous, Second International Conference or Data and Knowledge Systems for Manufacturing and Engineering, Gaithersburg Md, October 1989, available from the IEEE Computer Society, Press.

Hardware

The work performed for this task used the same hardware as that of Task 4.3.8.1

DICE PROGRAM

Task 4.3.3 Information Architecture Prototype GE-CRD

Summary Objectives

The objective of the DICE architecture team is to develop a generic integrating architecture "framework" that supports the "virtual tiger team" concept for concurrent engineering, is implementable within a reasonable timeframe, and above all is likely to find acceptance in today's US industry, subject to the current business pressures. The following architecture requirements support those over-arching requirements. The DICE architecture should:

1. Support concurrency for single users, cooperating users, and simultaneous users;
2. Provide shared access to a common information model, containing common schema, and new as well as existing data, files, records and databases;
3. Leverage existing "legacy" tools, data, databases and frameworks;
4. Accomodate the inevitable changes in standards, frameworks, tools and databases;
5. Allow tools, methods and other applications to run within the framework at speeds comparable to their independent execution;
6. Provide seamless access to the network-distributed users, systems and resources comprising the "virtual tiger team", *but within the context of the tools and/or environments which are familiar to the product developers*;

Requirements (1) and (2) support the virtual tiger team concept and are not inconsistent with the requirements for other integrating architectures. Requirements (3), (4), (5) and (6) are pragmatics, dictated by present economic, cultural, business and technical realities. Requirement (3) directly addresses the reduction of deployment costs for the architecture, as well as reducing the risk by permitting incremental deployment. Requirement (4), recognizes explicitly that many of the standards crucial to a concurrent engineering environment are evolving, inconsistent, likely to continue to change for some time, and moreover are not easy to influence. Taken together, this requirement puts the DICE architecture into an architecture class best described as open, public and above all, *pragmatic*. Most of the other integrating architecture efforts are based on the idea of fixing on a well-defined set of *idealized* standards and protocols. In these "idealized standards" architectures, single standards are proposed or adopted for all major interfaces. In most cases, unfortunately, the direct result of these choices are restrictions on the kinds of platforms, software, tools, and resources that can be integrated under the "idealized standards" architecture; as a result, these architectures are difficult, and extremely expensive to implement in "real" industrial organizations, with their legacy of existing environments with heterogeneous hardware, software and data. Moreover, unless they were designed explicitly with the expectation of changing requirements and standards, they tend not to survive changes without massive re-implementation costs. Finally, requirements (5) and (6) address useability and user acceptance, and are driven by the cultural reality that a concurrently engineering environment is peopled not by computer scientists, but mostly by engineers, who want to get on with their work.

The DICE architecture may best be described as an *application framework architecture*, (symbolic, but executable), rather than a *conceptual*, (high-level reference model, e.g.

CIM-OSA) architecture, or a *structural architecture*, (non-executable, structured descriptions), e.g. E-R models), or an *execution architecure* ,(concerned with machine-specific details, e.g. CISC/RISC instruction sets, floating point formats, etc). It is a new type of generic application framework, designed to integrate new *and existing* design applications, data bases, and frameworks.

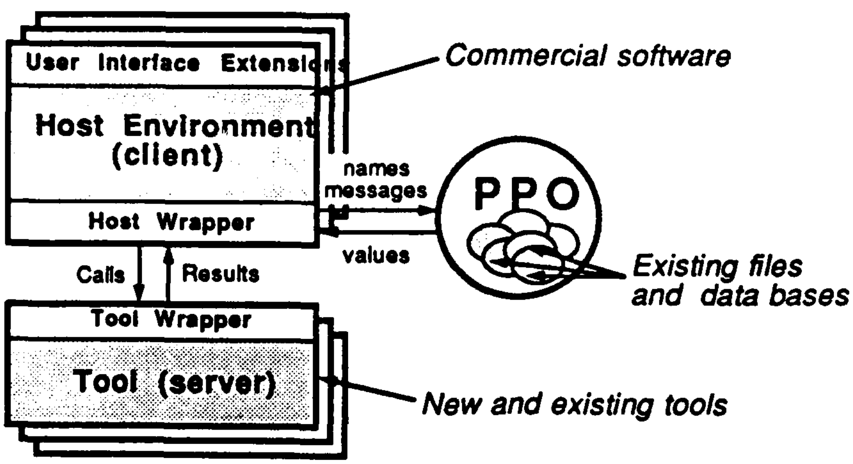
Overall Approach

In order to achieve a high degree of concurrency among product development groups in a large organization, the many applications and data bases which support those groups must be integrated. The approach now being implemented includes four basic elements: 1) Commercial host applications linked by wrappers or adaptors making tools, utilities, and data bases available as standardized system services, 2) flexible protocols for interchanging and sharing objects across multiple systems, 3) common, system-wide schema definitions for shared objects, and 4) a heterogeneous collection of data bases and files for permanent storage of those objects.

The approach is architecture–neutral, i.e., within relatively broad theoretical and practical constraints, each major group within an organization implements a product development environment using the product development tools and application frameworks appropriate to its computing environment and organizational structure. Then, *when integrated, the legacy tools and services appear to be native to the application framework.*

Clients, Servers, and Brokers

The Client-Server-Broker approach is based on the concept of using the "favorite" *commercial* applications of existing environments as "host applications" in which the DICE architecture services and environment are made transparently available in a manner familiar and comfortable to the end user. A diagram illustrating the concept can be seen in the figure below. In the diagram, commercial and "legacy" code and data are shaded, and the DICE achitecture can be seen as the "glue" that integrates, simplifies, coodinates and *adds value* to those existing (familiar and useful) services.



Most modern applications have six layers of functionality, which can be exploited as a basis for inter-operability and inter-connectivity. Although they are not standardized at present, standards bodies are slowly groping toward commonality. For purposes of

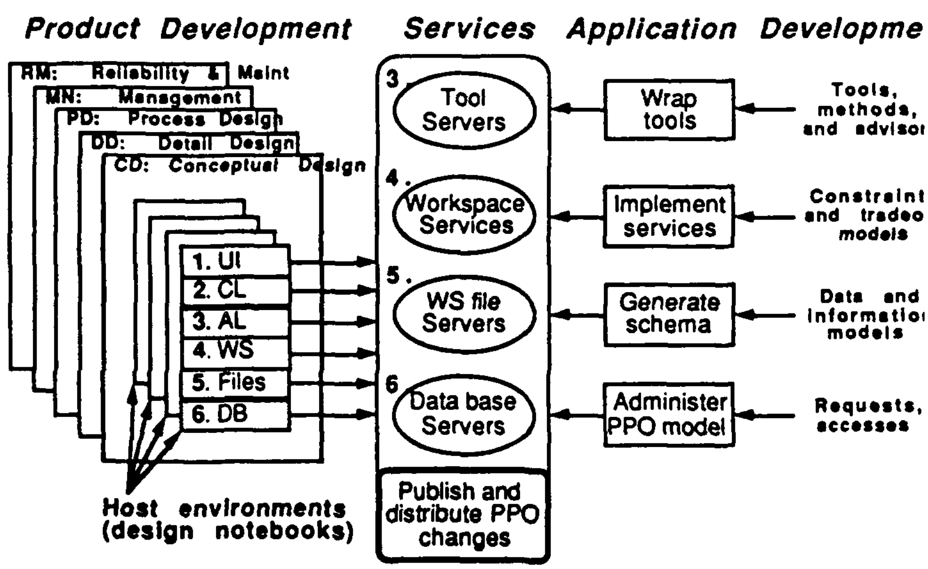
integration, the application frameworks, application environments, and applications themselves can be specified in terms of these six layers of functionality. Most applications architectures seem to be converging on some variation of the following six layer model.

1. *User interface:* External mechanisms enabling the user to execute functions provided by the system and to view the results. The current emerging standard appears to be some variation of the X-window graphic windowing system[1] with OSF/MOTIF libraries[2].
2. *Command language:* Language and interpreter for defining characteristics of the user interface, capturing sequences of user commands, and executing sequences of functions provided by the application. Long term directions here are not especially clear and will take some time to sort out. The Macintosh™ HyperCard language HyperTalk[3] is a good representative of this class of languages.
3. *Application language:* Language and compiler (usually) for defining the functions provided by the application. While FORTRAN is still popular in mechanical design, the C language is dominating many new applications and the C++ object oriented extension to C appears to be the choice of many electronics CAD system vendors[4].
4. *Memory manager:* Run-time subsystem for creating and manipulating objects in memory for use by the application language. Object oriented data base extensions of the memory object managers of languages like C++ seem to be most popular. Hopefully, system vendors will eventually provide language-independent object management systems and shared access to common object pools by independent application processes. Because most existing object managers do not save enough class information for run-time dynamic objects, some sort of run time object manager will also be required. Object oriented data bases such as ROSE can provide such layers[5]. Because of considerable technical complexity, standards will be some time in emerging for this area.
5. *Object interchange:* Run-time subsystem for saving and restoring collections of memory objects from files or sharing objects among invocations of the same or different applications in the same or different languages. Again object oriented data bases such as ROSE can provide such facilities through class libraries in languages such as C++, Objective C, and LISP/CLOS. Defining common semantics and schema for the objects still seems difficult, but the PDES/Express language shows considerable promise. With a number of viable options emerging, some standardization in this area seems likely in the next few years.
6. *Data management:* Utilities for managing changing collections of objects, files, and records on permanent storage (disk). Relational data bases have been standardized by ANSI[6]. Object oriented oriented disk data bases (the disk based equivalent of the memory management systems mentioned in layers 4-5 above) combined with iconic desktop user interfaces seem to be the most likely eventual technique for configuration management of files. A number of new data base products are emerging and existing relational data base vendors are extending their products with object oriented facilities such as clustering and rules. Still, an early decision in this area seems unlikely.

Facility	Description	Examples and Tools

1. User Interface	Mechanism for user to activate and monitor application facilities	Text, menus, X-Window, Open Look
2. Command Language	Language for defining user interfaces and capturing command sequences	UNIX shell, HyperTalk
3. Application Language	Language for expressing and executing application functions	FORTRAN, C, ADA, C++, Objective C
4. Memory Management	Data representation and manipulation services for in-memory objects	ROSE, C++, Objective C
5. Object Interchange	File formats and libraries for interchanging clusters of memory objects	IGES, PDES interchange, ROSE
6. Data Management	Language facilities, libraries, and utilities for managing the changing files, records, and other objects in permanent storage	UNIX file system, configuration control, relational data bases

In the DICE architecture, *Host Environments*, suitably *wrapped* via *gateways* connecting one or more of the *functional layers*, share *common servers*, providing access to the DICE architecture and the suite of tools, tasks, schema, objects, data and files that comprise a functional CE system. This view is illustrated in the diagram shown below, which further highlights the differing roles played by product and application developers in the DICE architecture.



In a client-server approach, the major application facilities are gathered into a collection of server processes called from the client through some form of transport layer. Each of the six layers has a different collection of transport layers and interfaces. Depending on the transport layer, clients and servers can reside on different machines with totally different execution architectures

1. *User interface:* In the X-window graphics system, the client process on a processor (not necessarily a graphic workstation) communicates with an X-terminal or a server process on a workstation via local or distributed character streams. .
2. *Command language:* Through character streams implemented by some sort of InterProcess Communication (IPC) protocol the client process sends text commands to a command line application server and receives results through the character stream.
3. *Application language:* The client application sees the remote application procedure as a local procedure call:
   - Linked into the client application at load-time,
   - Loaded into the client application at run-time,
   - Linked dynamically to an associated process at run-time,
   - Called through a token stream through a transport layer (Remote Procedure Call or RPC).
4. *Memory management:* Remote or shared memory can always be accessed through procedure calls, but transparent access requires an object-oriented programming environment.
5. *Object interchange:* The client running in the memory space of the application reads and writes files through a server which tracks changes.
6. *Data management:* Clients running in one or more distributed computers can call a central data base server through a remote procedure call implemented over any of the network transport layers.

PPO Model Management

Implied in each of the six layers of functionality are models for the objects involved and their behavior. Generally these models are implicit in the specifications and implementations of the languages, data bases, interchange formats, and applications in each of the six layers. Characterizing and standardizing these models is an important first step to integration if it can be accomplished in a timely manner. Of particular importance for concurrent engineering are unified parametric and pseudo-parametric models for the Product (form and function), the Process (activities involved in design, production, and support), and the Organization (resources of all kinds). These PPO models enable product developers to understand the interactions of customer requirements, performance, cost, and supportability parameters before the parts, pilot processes, physical models, or detailed computer models have been constructed. Moreover, if plans fail or requirements change, these models enable product developers to rework these tradeoffs within design, product, and support constraints.

Obviously, a single monolithic PPO model would simplify system and applications development. Perhaps, some time in the next ten years, a sufficiently powerful data definition scheme will arise; standardized data bases implementing that scheme will mature; the generic schema for business areas such as aerospace structures will emerge; individual organizations will define the specific schema required for their operations; vendors will rewrite their codes to match this data base; application developers within the organization will embrace the new scheme; all of the product development organizations

will populate that model; and the organization will buy enough compute power to run it all efficiently.

More likely, for some time the PPO will remain a heterogeneous collection of flat files, disk data bases, and object oriented data bases of various sizes and degrees of maturity. A concurrent engineering system must manage changing versions of the files and data bases representing the detail models as well as the processors for extracting descriptive parameters from those files or generating the detail models from those parameters. Consequently, a practical PPO model can be expected to have two interconnected components :

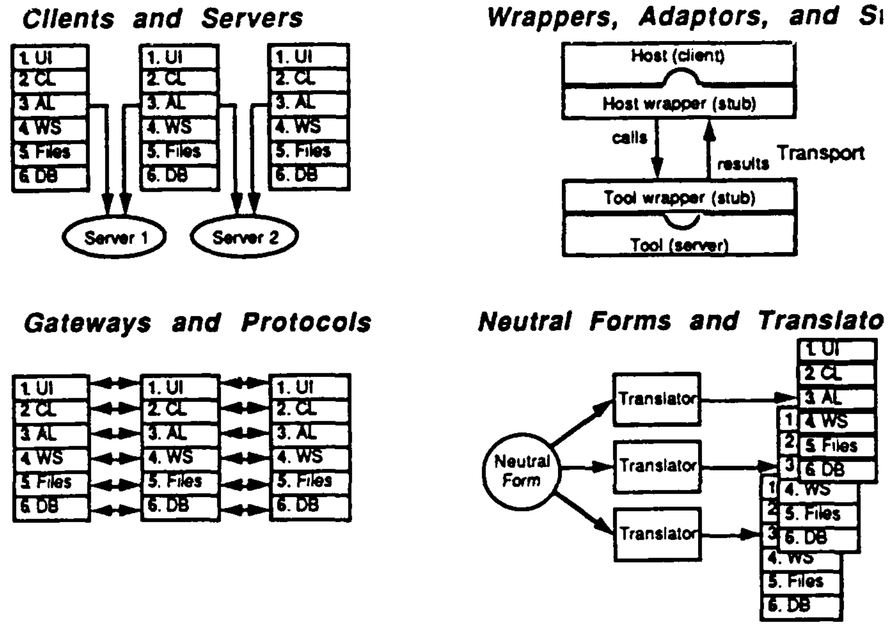
- The detailed descriptions of each perspective on the PPO (now captured in the files or data bases used by the relevant processors), and
- An auxiliary model which relates key parameters in multiple perspectives through tradeoffs and constraints (probably captured in compatible object-oriented data bases).

In a typical product development environment PPO models exist in multiple forms in the six functional layers of an application. Of most importance are the memory resident objects, interchange files, and data base objects. In some situations, formal data definitions or schema may exist, but more frequently the interpretation of this data is left to detailed application code and without this code, the data has little meaning. The major objective of PPO modeling is to define the semantics of the PPO model independent of any particular application program which uses that data; in particular

- To define parametric or approximately parametric models for interrelationships of key constraints and tradeoffs
- To maintain a unified schema for those models and their interrelationships with the detailed product and process models, and
- To publish and distribute changes to those models among the different representations and platforms.

Integration Techniques

Given a collection of disparate tools, services, and frameworks, there are four basic mechanisms for achieving some degree of interoperability. Ideally, each tool or service is written in the most convenient environment for the author and that service is mapped into every other environment, where it appears as if written especially for that environment. Similariy data is defined and managed in the environment most convenient for the owners of that data, then appears in each host environment as if defined originally in that environment. Four basic schemes for integrating applications and data bases, (client-server, wrappers, gateways, and neutral forms), are shown in the figure below.



Wrappers, Adaptors, and Stubs

Many applications cannot be accessed directly as servers because of their form or calling sequence does not match the clients' conventions. One alternative is to rewrite each method, tool, or advisor to match the calling conventions and data representations of the client. The result is a maintenance headache with multiple implementations of equivalent codes and the inability to move applications to a new environments as needs change. Wrappers, adaptors, and stubs are interface modules which enable the applications to be written once and subsequently called from multiple heterogeneous clients. Already Remote Procedure Calls (RPC's) give a limited version of this facility to applications which support them[7]

Wrappers typically come in pairs. The host wrapper is resident in the host application or client. It is composed of a host broker containing tables of the external services and a collection of interfaces to the different interface protocols implemented (for example, the four calling sequences for procedures listed in the previous section). Many host environments can implement external functions which the user cannot distinguish from those written in the host application's extension language or embedded in the application. When called, the name of the external function is found by the broker, arguments are unpacked and reformatted by the interface code, and the external procedure called through the appropriate transport layer.

Tool wrappers are the interface modules which connect existing procedures, libraries, and applications to the remote call protocols or procedure interfaces which make their services available to other applications. A tool wrapper unpacks the arguments and converts the call from the transport layer into a call compatible with the tool.

Gateways and Protocols

Gateways are interfaces among different host environments at the same functional level. Many of the protocols required to implement gateways now exist. For user interfaces,

tokens can be exchanged among X-servers; for command languages, command character strings can be exchanged through queued messaging services like the DICE Local Concurency Manager (LCM)[8]; for procedure calls there are the remote procedure calls discussed earlier; and for files and processes; there are the object oriented data bases discussed earlier.

The major barrier to this integration technique is the requirement for a dramatic change in host application design. Each application is actually going to need a scheduler in order to handle the concurrent event streams from the user, subordinate processes, and other cooperating applications. For example, if a user were in the process of building a drawing on a CAD system, that user might not appreciate another user updating the library or changing the geometry at the same time. A combination of computer and people-people protocols are required to coordinate these activities.

Neutral Forms and Translators

The final approach to integration is the introduction of standardized neutral forms, languages, and interchange formats. Programming language standardization has been relatively successful. FORTRAN, C, COBOL, and ADA languages have extensive formal specifications and carefully written verification suites. Moreover, programs which survive processors such as LINT[9] are likely to run on nearly any C language compiler. Many vendors provide helpful extensions to both C and FORTRAN, but organizations which resist the temptation to exploit these extensions can run their codes on most platforms without change.

The situation in exchanging CAD models or product/process models among different groups is much more difficult. The difficulties arise at four different levels:

- *Function: Do the entities actually represent the same object?* In preliminary design, the part might be a lumped mass and spring model with no geometry whatsoever. In aerodynamic design, the geometry is for stressed and extended object at full operating temperature. In manufacturing, the mold or fixture geometry is corrected for tooling wear, shrinkage, and other process effects.
- *Realization:* Are the entities being represented in the same way? For example, a rounded corner might be realized as either a spline or as a circular arc. Both realizations do essentially the same thing, but reliable conversions between the two representations are not automatic, especially if the conversion program does not know a priori the both are implementing a rounded corner.
- *Representation:* Splines and circular arcs can be represented by a variety of different schemes ranging from rational polynomials to B-splines. Some are equivalent, but many cannot be converted automatically in general. Roundoff error is a persistent problem[10].
- *Encoding:* Finally, even if the representations are equivalent, they might be encoded differently. Number representations and data structures vary. While the conversions here are generally automatic, they do require time and roundoff error can lead to problems again.

Implementation

Gradually over the next five to ten years, most applications will be rewritten to conform to one or more of the emerging applications architectures. The application software market will gradually change from the current vertically integrated marketplace (by application

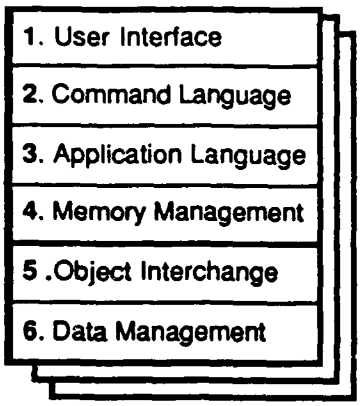
domain) toward a horizontally integrated (functional) marketplace of frameworks, host environments, and servers of various kinds. Many of the tools and services will be object oriented so that mixing functions will be more straightforward. The challenge to product development organizations is dealing expioiting these changes without discontinuities in operations or excessive cost and risk.

Restructuring and Reorganization

The first task is a gradual restructuring and reorganization of existing codes. Integration is easier if applications can be reduced to one of four different categories described below:

- Host environments (H): Open and extensible applications implementing a significant portion of the six basic functions and integrated into a common data base.
- Procedures (P): System-independent procedures free of input/output calls;
- Filters (F): Programs which read some collection of files (or character streams) and produce another collection of files (or character streams)
- Command line interactive (C): Programs which accept text command strings

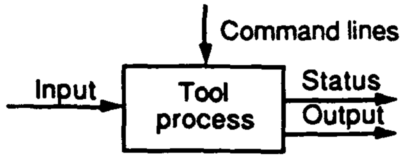
Type H. Host environment



Type P. Procedure tools

```
double pmr_ilss( fiber_fraction, void_fraction)
    float      fiber_fraction, void_fraction;
{   float      ilss;
    ilss = . . . ;
    return(ilss);    }
```

Type C. Command Language Tools



Type F. Filter tools

Technical Results

An implementable architecture for CE has been designed and prototyped; details of technical results can be seen in the individual sections following this introduction. In summary, however,initial results include several different spreadsheet and mathematical workbook host environments with wrappers for complex optimization codes and supporting functions in C and FORTRAN. Communications protocols supported include the DICE LCM, DECNET, and TCP/IP. Plans for future host environments by other groups on the program include LISP/CLOS, relational data bases, and the Laser expert system shell. Preliminary PPO models and configuration control systems have been constructed for the model problems addressed by the CE Testbed, along with a functional suite of tools allowing rapid schema modeling, followed by automated creation of the schema,directory structures required for use with target databases. The prototype system works well, and meets all system requirements tested thus far.

Conclusions

The approach described above seems to be sound, and no serious technical obstacles have been discovered in the course of technical development and implementation to date. .

The approach seems to have strong appeal for corporations with significant "legacy" code, environments and systems, and who wish to pursue CE technology on the basis of incremental deployment. While the six-layer applications frameworks now emerging will provide the required services, most organizations will be supporting both existing code and the new architectures concurrently for the foreseeable future. By applying a careful mix of host environments, servers, gateways, and heterogeneous PPO data management, organizations should be able to keep product development environments running effectively

Recommendations

Continue development, with additional emphasis on field trials, and finding and solving the problems associated with interfacing to legacy systems.

---

[1]*X11-Windows Specification.* X-Consortium. Cambridge, MA, 1989.

[2]*MOTIF.* Open Software Foundation, Cambridge, MA, 1989.

[3]*HyperCard Users Manual.* Apple Computer, 1989.

[4]B. Stroustrop. **C++ Programming.**Addison Wesley, Reading, 1987.

[5]M. Hardwick. *The ROSE-IC Library for Objective C: Persistent Objects for Objective C*, Tech. Report, Computer Science Department, Rensselaer Polytechnic Institute, Troy, NY, 1989.

[6]Data Definition and Query Languages for Relational Data Bases. American National Standards Institute, New York, New York.

[7]*Network Computing System.* HP/Apollo, Chelmsford, MA.

[8]R. Kannan, K.J. Cleetus, and R.Reddy. *Distributed Computing with Concurrency Manager, Proceedings of the Second National Symposium on Concurrent Engineering*, Concurrent Engineering Research Center, Morgantown, WV, February, 1990.

[9]*LINT.* Unix Programmers Guide. ATT.

[10]J.W. Lewis. *Interchanging Spline Curves Using IGES.* **Computer Aided Design 13** (1981) 359-364.

Task 4.3.3.1 VMS Thread Components

Objectives

The objectives of this task are to develop VMS software components supporting the DICE CE environment, and the Phase II demonstration.

Approach

The overall development plan is to carry out most of the generic software development in Ultrix, and to port completed components to the VMS environment.

Technical Results

Since it appears that many of the UNIX services which Ultrix provides, (and which the DICE development leverages), will soon be available under VMS, little effort was spent in re-creating those services. Resources were expended however in providing interfaces, and translators and in general integrating the VMS based manufacturing tool suite, specifically IDEAS, TRUCE, and NC Verify.

Conclusions

Many necessary standard services for full, reliable inter-connectivity and inter-operability are not yet available under VMS, and are appearing very slowly; this limits alpha site selection.

Recommendations

Waiting for UNIX-compatible supported standards is no longer practical, since one of the Phase II alpha sites demand some degree of VMS/UNIX data interchange capability. A limited effort should be launched to provide limited data interchange capability consistent with anticipated commercial underpinnings. The VMS effort should be small, and should make use of the "Gateway protocols model" described above for application inter-operability and inter-connectivity.

Task 4.3.3.2 UNIX Thread Components

Objectives

The objectives of this task are to develop UNIX software components supporting the DICE CE environment, and the Phase I demonstration.

Approach

The overall development plan is to carry out most of the generic software development in Ultrix, and to port completed components to other UNIX environments.

Technical Results

Many Ultrix code components were developed/modified/integrated as part of the Phase II effort; functional descriptions appear later in this document in the those sections where the components are integrated to provide overall functionality.

Specific codes developed/modified/integrated/improved are:

- XS spreadsheet tool
- Q-Calc spreadsheet tool
- TAE+ outer wrapper for XS
- Framemaker-based formal EDN (Electronic Design Notebook)
- Data wrappers for
    - ADS optimization code
    - ROSE database code
    - BEAM namelist FORTRAN program
- Change Management System prototype, based on SCCS, and extensions, developed and demonstrated.
- "genSchema" automated schema generation tool
- "update_formats" tool to create instances in th six supported formats
- Directory Builder for ROSE schema
-

Conclusions

The development, modification and/or integration of these modules went smoothly in a uniform, common single operating system environment, however when the software was transitioned to CERC inconsistencies were discovered during integration with modules developed elsewhere under the program.

Recommendations

A rigorous set of standards, interfaces, protocols, procedures and documentation should be set up and enforced DICE-wide across the program, and focused on the "rocks" that make up the DICE CE Testbed.

Task 4.3.3.3 Workstation Node Prototype

Objectives

Develop and/or integrate the basic technical underpinnings needed by a user's application programs running on a workstation platform to "connect into" the DICE information architecture framework; these basic services consist of:

- "wrappers" whose function is to encapsulate the applications that the user wishes to run, providing inter-connectivity, inter-operability and transparent access to the global object space, and
- the ROSE "client" process, whose function is to provide version-controlled access to persistent global objects.

Approach

The work associated with this task was initiated in of a series of studies to define and constrain the problem and to determine the strengths, weaknesses and overall applicability of existing software tools to achieve the desired functionality. Next, modules providing the needed functionality were acquired, and/or developed through rapid-prototyping. Finally, these modules were integrated to provide a suite of existing old applications with the desired new properties of inter-connectivity, inter-operability, shared global object space and object persistence.

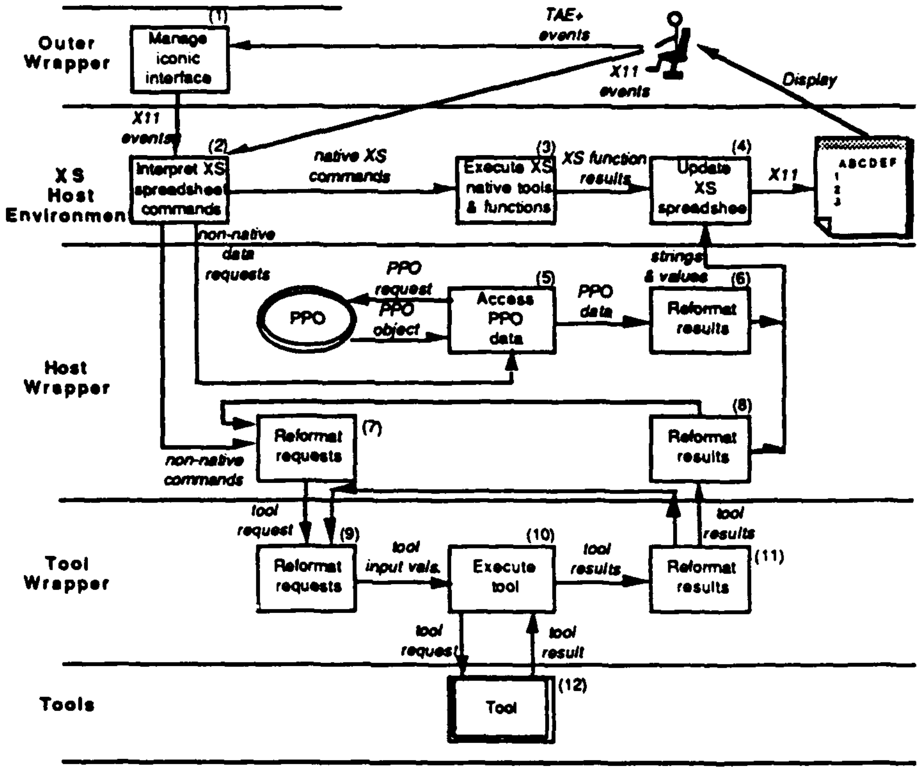
Technical Results

Module Development

XS Integration Environment

A prototype of the wrapper concepts described to this point have been implemented as part of the DICE Concurrent Engineering Demo. The prototype wrapper concept was initially developed for a spreadsheet integration environment, using an X-Windows based extensible Lotus 1-2-3 emulation called XS, and was (initially) targeted for Ultrix systems. The integration environment includes a spreadsheet (XS) with access to PPO data, an optimization package (ADS), a FORTRAN namelist oriented method and DICE kernel services. The spreadsheet interface is used to reference PPO data, experiment with data, and optionally publish results. User experimentation can include conventional spreadsheet operations, custom XS functions, namelist methods, and both optimizations and parameter studies with dynamic display of intermediate results.

This integrated environment consists of several applications packages integrated and coordinated by their corresponding wrappers. A host wrapper allows XS (a Lotus-compatible, X-windows based, spreadsheet) to be the user's view into the system. An outer wrapper allows the user to optionally access XS commands from a menu-oriented interface. Separate tool wrappers allow access to beam stiffness (namelist FORTRAN) routines, to optimization routines, and to parameter study routines that specify loop control for the selected namelist method.



XS Integration Environment Dataflow

The constituent components of the XS Wrapper environment function as described below:

(1) translates TAE Plus menu and icon selection events into X events which are understood by (2). As such, it "outer wraps" the host application's Lotus 1-2-3 style keystroke interface with a TAE Plus point and click user interface.
(2) provides XS with an X11-based user interface. User entered and calculated data may be sent to intrinsic and user defined functions as individual cells or ranges of cells. It is the host environment from which the user sees all wrappedapplications and utilities.
(3) executes spreadsheet functions (such as SIN, AVG and SUM) and other "native" tools and functions which have been added to the spreadsheet. It pops range, value and string arguments off parameter stacks, calls the specified function, handles errors and returns values. (3) also resolves cell dependencies.
(4) Formats string, integer and real data to be displayed in individual cells of the spreadsheet.
(5) uses DICE kernal services to access data objects from the PPO. It uses configuration managements routines to check applications objects in and out of PPO directories before they are read using ROSE.
(7) converts cell ranges in a pre-defined format into a set of string and real arrays to be used external to the host environment.
(6) and (8) convert external data (e.g., PPO information and method results, respectively), respectively, into predefined, tabular format for viewing in the spreadsheet. Functions (5-

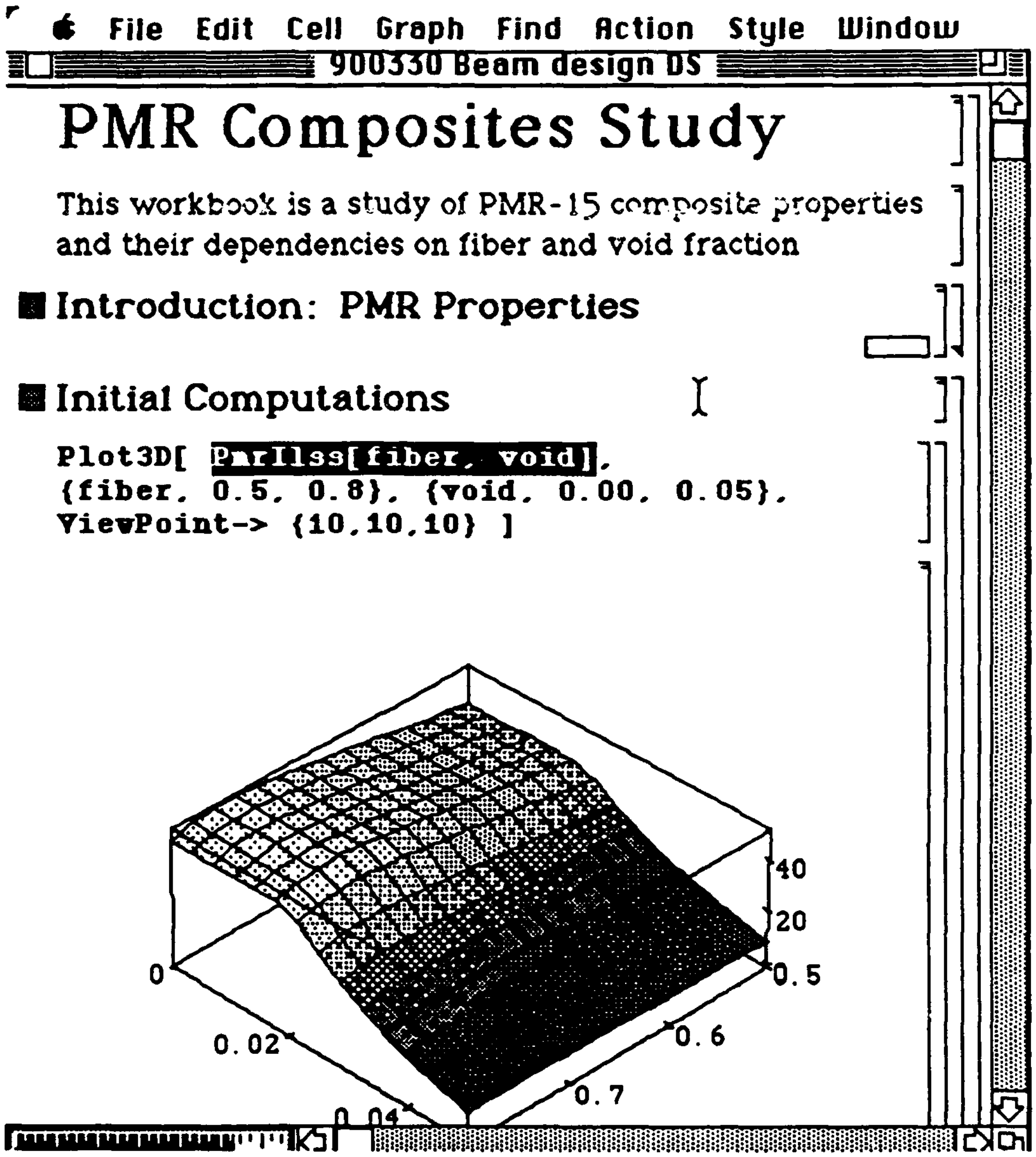
8) together form the host wrapper which translates data to and from the form it appears in the host environment.
(10) manages the execution of a tool, which may be by procedure call, subprocess invocation, or remote execution using LCM.
(12) is an external tool. The two types of tools currently being wrapped are (a) a FORTRAN namelist method, or (b) an "iteration tool" which computes a set of input values to be used in the next execution of a method code. In (b) the values may be calculated based on previous results and optimization heuristics, or based on pre-set value increments (parameter study).
(9) and (11) perform the tool wrapper's data translations. In the case of (12a), since these methods use namelist I/O, (9) and (11) translate data to and from FORTRAN namelist format.

This XS Wrapper environment has the following interfaces to the outside world:

- TAE Plus "outer wrapper" provides an Excel-like menu, and icon user interface which sends X events to XS spreadsheet window, supplementing XS's Lotus 1-2-3 style keystroke interface, and *providing access to the rest of the DICE services from within the XS "host applications environment"*.
- Access to the PPO is through direct ROSE invocations from the Host wrapper, not from either the tools or the XS spreadsheet directly.
- A file based persistent object protocol implements hot-links of value updates from the spreadsheet to the Project leads host environment (WINGZ and the Macintosh).
- Methods are executed locally by subprocess invocation.
- XS worksheet files may be read or written at any time. This effectively sa es a users current analysis data at any given time.
- Tools can be imported via a tools server. E.G., The BEAM (namelist) wrapper tool will automatically read and write the namelist files, and export the results to the XS spreadsheet.
- XS spreadsheet cells is also "hot linked" to send data to the other nodes, e.g. the information manager node.

Mathematica EDN

n the December demonstration, the symbolic mathematics tool *Mathematica* was used as the host application for an engineer's notebook. Mathmatica is a notebook-oriented tool, which in addition to impressive capabilities, (calculator, graphics engine, symbolic math interpreter, programming environment, and notebook), it has the hooks and handles to run external applications; in particular, Mathemnatica support a fairly rich subset of (UNIX) services which make it an excellent candidate for a DICE "host application". It runs on multiple platforms and supports applications running on other (multiple) platforms. A minimal "Engineer's Notebook" level prototype EDN was developed for this demonstration. The organizational model was based upon a scenario drawn from actual design usage at General Electric Aircraft Engine Division. A sample of the Mathematica EDN is is presented in the figure below. The function **PmrIlss** referred to in "Initial Computations" in the figure is in fact actually evaluated on a remote tool server, the call in Mathematica looks as if it were part of the supported Mathematica function library.



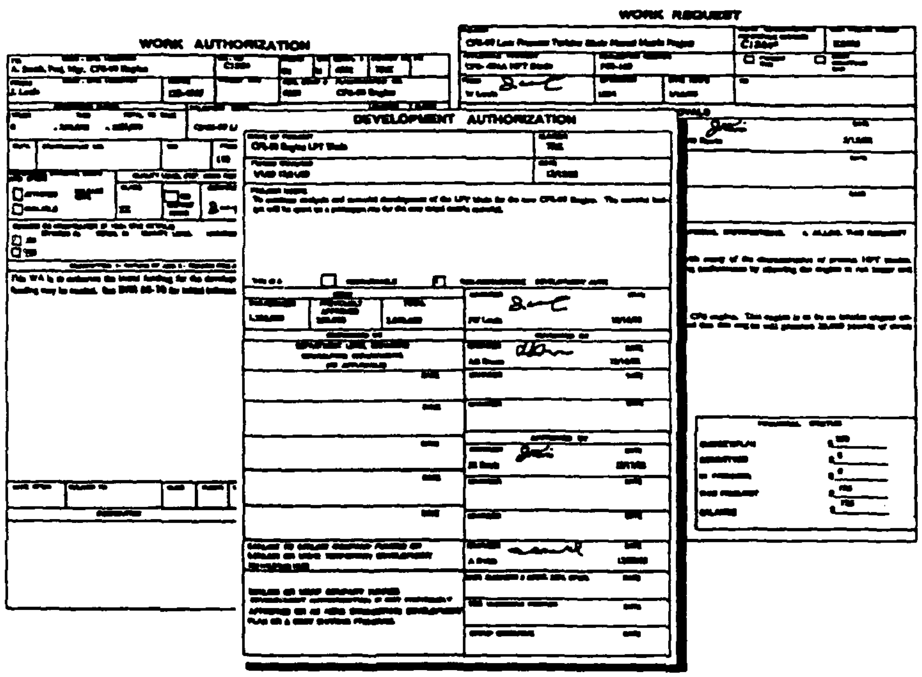
Mathematica EDN

The Mathematica-based EDN was flexible, easy-to-use, and provided adequate support for the "host environment" role; specifically, it supported the engineer's needs for a unified working environment through:

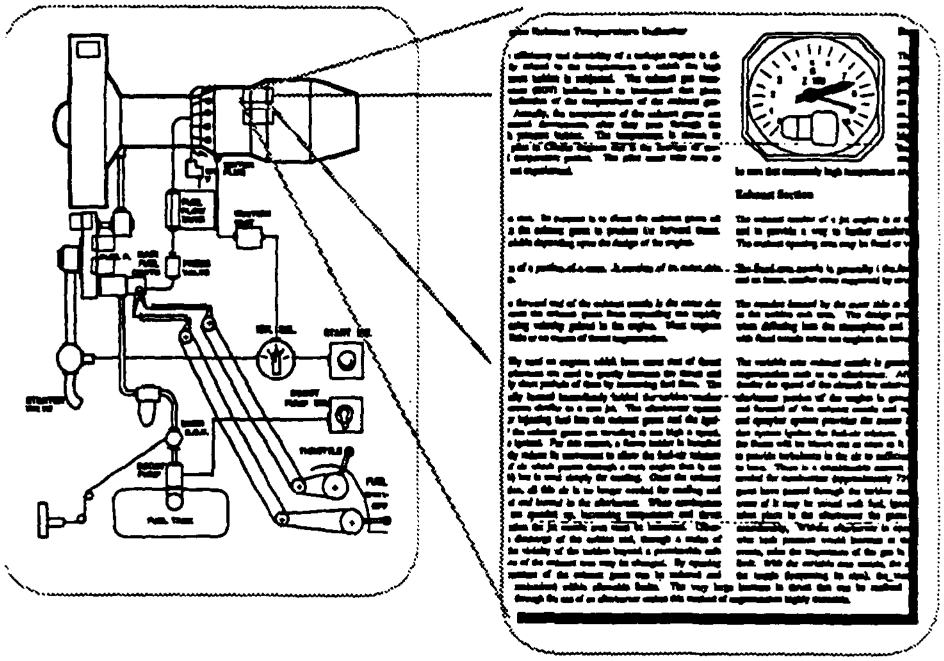
- unified consistent interface to heterogeneous tools
- invocation of remote functions
- storage and retrieval for the document through the PPO

Framemaker EDN

In the December demonstration, the text editing system FrameMaker was used. FrameMaker is X Window based desktop publishing system. It is primarily a WYSIWIG text editor with graphics, image and hypertext capabilities. It runs on multiple platforms and the data was easily moved and used on multiple platforms. A minimal "Patent/Project Notebook" level prototype EDN was developed for this demonstration. The organizational model was based upon General Electric Aircraft Engine Division's Design Record Book (DRB) which is made up of Design Summary Studies (DSS). The basic form of the DSS is presented in the figure below. One of the nice features of this approach was the ability to bring in graphic data from other applications, including IGES graphics and bitmapped images.



Framemaker EDN "forms"



Framemaker EDN "hypertext"

Because of its background, FrameMaker did very well in the area of presentation . The current version of Framemaker does not support dynamic change or linkage to other applications. As a result, data and commands had to go through filter programs outside of FrameMaker with no logical connection to FrameMaker. It is anticipated that future releases will support external hooks, so that his lack of "intimacy" with other applications should be correctable.

In the Phase II work, the link support was done by using filters, a small database, and the use of local links (link information is kept in the node to speedup response). The current FrameMaker interface in the creation of links is cumbersome and not intuitive. The use of filters meant a delay in creating all the links of the EDN. This was perhaps acceptable in the Patent/Project Notebook level but completely unacceptable in more informal notebook concepts, such as the Pad of Paper level or Lab Notebook level.

Conclusions

On the basis of experiences in developing and demonstrating these concepts, several conclusions were drawn.

- The spreadsheet paradigm for users is a familiar one for development engineers; they are comfortable with the interface, and the programming concepts provided. Extensions of these programming facilities to include invocation of external applications, and global persistent storage was very natural, and well accepted by the user community. In general, the "Macintosh"-style menu/icon interfaces employed were viewed somewhat suspiciously as "gimmicky" by line engineers.
- The wrapper concepts described above were relatively simple to implement; they appear to offer the potential for cost-effective generation of wrappers for the application classes encountered in the demonstration. They support adequately the overall system requirements of inter-connectivity and inter-operability.
- The "Framemaker"-based EDN was capable of producing first-class formal documentation , but suffered from weaknesses in the vendor's support for links and external interfaces. It is anticipated that future version will correct this problem, and provide the basis for a more robust EDN.

Recommendations

In light of the development and demonstration experiences and conclusions, several recommendations are made:

- Continue development of the wrapper concepts described above
- Extend the spreadsheet integration paradigm, to further leverage end-user familiarity with spreadsheets.
- Wrap additional examples of current application classes, as well as new ones; verify the cost-effectiveness of the wrapper concept by demonstrating auto-wrapping for some classes. Generalize the approach.
- Generalize the global object access methods.
- Complete linking the EDN directly to the PPO to provide support for versioned design notebooks.
- Investigate other EDN host applications

Task 3.3.3.4 Fileserver Node Prototype

Objectives

To enable product development groups to achieve a high degree of concurrency, a shared model of the Product (both form and function), Process (activities in all life cycle phases), and Organization (resources of all types) is required. Eventually these PPO models might be stored using a uniform set of engineering models in a uniform representation scheme and a common database, but that degree of standardization is certainly impractical for most organizations in the next ten years. Consequently, DICE PPO services are implemented as a heterogeneous collection of files, relational databases, languages, and object oriented databases. DICE must provide an information system which supports this heterogeneity.

Commonality is achieved in this heterogeneous environment in several ways, the most fundamental of which is through a common, system-wide schema. The schema is defined using an "information pipeline" in which raw information is first collected as an informal scrapbook, a formal structured model is then created, and finally a data definition leads directly to an implementation. Ultimately, the DICE CE data model must be able to accommodate *dynamic changes* in the PPO schema which emerge from this pipeline.

Approach

In a distributed information system, there must be integration at the data level in order to allow distributed users and applications to communicate and cooperate in a meaningful way. If a system were designed from scratch, data structures could be designed and used consistently throughout the system.
Unfortunately, DICE does not have this luxury. In order for DICE to succeed, the vast collection of existing heterogeneous engineering applications must be retained and integrated into the DICE environment. Replacing the many applications which have developed over a period of many years would not be a practical solution.
The approach taken by DICE is therefore to develop a uniform organization of existing data structures and formats, in order to provide tool integration to the extent possible for existing tools. While this will not lead to perfect, *tightly-coupled* tool integration, it is proposed that considerable benefit will still be achieved by integrating tools in a *loosely-coupled* way.
Commonality is achieved through the use of

- a common, system-wide *schema* (information model definition),
- a common directory structure, based on the schema, for organizing files of all types,
- structured files, also based on the schema, which contain important parameters and which reference tool-specific files, and
- a common configuration management system.

The PPO schema defines an information model describing the product (form and function), process (activities of all sorts), and organization (resources of all kinds). This schema specifies both the attributes and behavior of all shared objects system-wide. The structured files support a heterogeneous set of interfaces. PPO objects must be accessed as objects in a number of languages including C++ and Objective-C, CLOS and Laser, (although in Phase I only objective-C was supported). In the Phase I prototype, the experimental ROSE object-oriented database system from RPI, together with some specialized DICE utilities, support this capability in Objective-C.

In addition to the *core model* which replaces the blueprint in defining the product in detail, the PPO should contain a collection of *auxiliary models* which:

- convey design intent downstream to production and support,
- convey constraints and tradeoffs upstream from production and support to design, and
- communicate anticipated progress and response to change among groups working concurrently.

Theoretically, a complete model of the entire manufacturing enterprise could be generated and exploited for concurrent engineering. In particular, for simple, stable products, both the core product definition and auxiliary models can be captured as parametric models. *For more complex products and organizations, the core models are too complex to be translated easily from the forms in which they are generated into integrated parametric databases.* In this case, the core models are left in their original forms (files or databases) and the principal relationships among design parameters captured through rules of thumb, linearization, data fitting, and reduced order models.

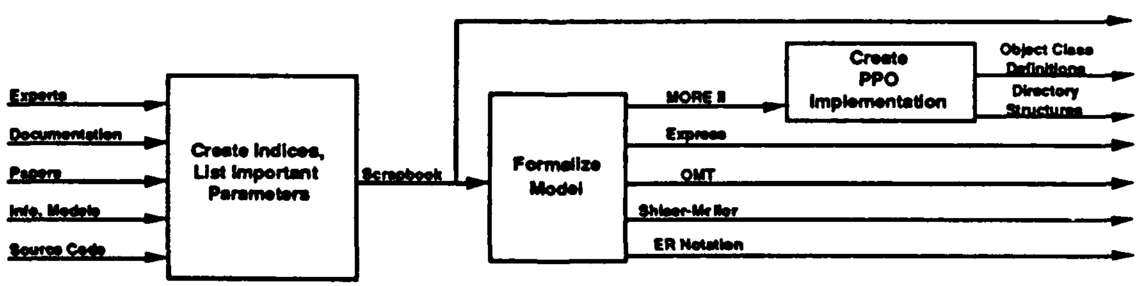
Common system-wide schema are defined using a pipeline in which raw information is first collected as an informal scrapbook, a formal structured model is then created, and finally a data definition leads directly to an implementation.

The task of generating and maintaining PPO schema breaks down into three concurrent processes. First, a scrapbook of programs, data definitions, papers, and other data is collected. Second, that scrapbook is structured using data flows, entity-relationship diagrams, and other structured analysis tools. Finally, the structured document is converted into a computer-executable specification from which schema for the various languages and databases can be generated.

Technical Results

Information Pipeline

An "information pipeline" was assembled, in which raw information is first collected as an informal scrapbook, a formal structured model is then created, and finally a data definition leads directly to an implementation. This was used to assemble the PPO model used in the December demonstrations.



As can be seen in the figure above, a thin, but functional thread through the pipeline was put together, and used to produce the class structures and directory structures used in the December demonstrations. The initial implementation of the PPO model includes specific data on a particular aircraft engine parametrization, general project management, and simple tables for capturing constraints and tradeoffs. The schema for the model are defined using a simple outline processor. The model itself is stored as a collection of files stored in

directories whose structure reflects object relationships within the model, as defined by the schema. Multiple views are implemented using multiple directories and links to files. The files are heterogeneous and include tool-specific files and also structured ROSE files. The following sections describes the methodology which we have developed for maintaining our heterogeneous information system for CE.

PPO Scrapbook

In Phase II, work on the scrapbook continued, consistent with producing an information model complete enough to serve as a representative model for the concurrent engineering demonstration in December 1989, as well as for the CE Testbed at CERC. As before, the mechanism chosen for reducing the time and labor involved in data model creation was to start with existing documentation. The term "scrapbook" is used to denote the fact that all available documentation and information is collected and organized into categories, without yet imposing structure or format on the information. Until all appropriate information is collected, the appropriate structure and organization may be difficult to determine.

For the CE structures testbed, a scrapbook was created which describes jet engines. Information was taken from various sources, including:

- textbooks on engine design,
- training documentation from GE Aircraft Engine Business Group,
- other documentation from GE AEBG,
- papers from West Virginia University, and
- conversations with experts at GE CRD.

The scrapbook contains diagrams and explanations of the jet engine form and function. The major sections are identified, as are the major design goals, such as low weight and high efficiency. The scrapbook attempts to capture the important concepts involved in creating the design. The characteristics of jet enginesfor example, are governed by a number of complex, interrelated equations and heuristics (rules-of-thumb). All relevant equations and design rules should be captured in the PPO Scrapbook.

These equations are based on many parameters whose values will ultimately be stored in the PPO. There is another dimension of complexity in that the values of these parameters depend in some cases on the operating condition of the engine. For example, efficiency may be a function of velocity, air pressure, temperature, and other factors. In fact, different designers may be concerned with essentially different models of the same part. The mechanical engineer may be interested in a blade which is at operating temperature and under stress, while the manufacturing engineer is concerned with the production and assembly of parts under static conditions. The characteristics of a part differ greatly depending on its temperature, stress loading, and so forth.

The scrapbook contains a set of pages which contain descriptions of attributes. The scrapbook may also contain photocopies of documentation which provides further explanation for an attribute. For the case where attribute values depend on dynamic operating conditions, it is appropriate to maintain tables of values. Also contained are such diagrammatic aids as design flow graphs and product assembly trees. The scrapbook therefore contains information relating to both the product and the process of a design. An *activity model* can be used to either guide (prescribe) or monitor (describe) the design or manufacturing process. Design activity models are appropriate for projects which focus on incremental design. Incremental design is where the general structures being designed

are well understood, and only incremental changes in parameter values need to be determined. For original design, it is much more difficult to create an activity model in advance. Jet engine design generally falls in the category of incremental design, and is therefore conducive to the use of activity models.

Data Modeling and Formal Representations

From the PPO Scrapbook, a formal information model was created. This information model must be unambiguous and internally consistent. In other words, the meaning of the information model should be clear, and not subject to individual interpretation. The model should also translate easily into different possible implementations, without being limited to any particular implementation.

The modularity between the data model formalism and any particular implementation will facilitate the simultaneous development of the information model and the implementation. Substantial effort was put into studying available modeling techniques and their applicability to this problem.

Product Models

Information modeling techniques have generally been derived as aids to database design. As such they tend to support the modeling of *entities*, and the *relationships* between entities. An entity is defined as being some thing which exists and is distinguishable and which has *attributes*. Relationships may also possess attributes which properly belong to the relationship itself, rather than to any of the associated entities. This terminology is general enough to be applied conceptually to any of the information modeling techniques.

The important point is that an information model should be a complete and unambiguous description which does not constrain the type of information system which will ultimately be used to implement the model (although entities, attributes, and relationships generally map quite naturally into relational databases).

Since products generally consist of tangible items, it is generally straightforward to create at least an initial model of a product. Intangible items, such as control, activities, states, and so forth are generally more difficult to model. The standards community has defined a variety of data transfer standards, and traditional (sequential) manufacturing and design work has developed models for specific products, which are in many cases proprietary.

Activity Models

There are many techniques for representing activities to be found in the modeling community, the AI community, traditional business-oriented project management techniques, the engineering design community, and so forth. These techniques may be concerned with the dynamic behavior of objects in real-time systems, or with the robust representation of project activities for the purpose of generating, scheduling, and chronicling activities.

Some modeling techniques provide specific support for representing the life-cycles and data flows of objects (i.e. Shlaer-Mellor notation), while other techniques provide very general modeling notation which allows any type of information to be modeled, but with no special constructs for modeling processes (i.e. E-R Notation).

Graphical vs. Text-Based Formalisms

Modeling techniques can be either text-based or graphical (diagrammatic). Formalisms of either type can fulfill the requirement of providing a consistent, unambiguous notation for representing information models.
Graphical formalisms are generally easy to comprehend on inspection, assuming that the conventions of the formalisms are understood. For example, in Shlaer-Mellor notation, an arc with a small cross line represents an inheritance relationship. Also, arbitrary networks of relationships can easily be created in a diagram.
Text-based formalisms tend to have the opposite problems and virtues. Existing word processing and outline processing software provide convenient editors for text-based notations, and are generally extremely fast to use. Text-based formalisms allow a very high density of information to fit on a page. However, arbitrary networks of relationships are difficult to represent using a text-based notation. Hierarchies can be easily represented in an outline form, but complex networks can only be represented in a rather clumsy way by using multiple hierarchies.

Modeling Tools
The remainder of this section lists the available set of data modeling tools evaluated for use in DICE, and describes briefly the tools actually used for schema modeling in Phase I. MORE II, PDES/Express, ER Notation, Shlaer-Mellor Notation, Object Modeling Technique, and Logic-Based Modeling were all studied as potential schema creation tools. MORE II and PDES/Express were chosen as the initial vehicles for schema modeling under DICE, will be described in more detail.

MORE II
MORE II is the tool which has been chosen for the *initial* data modeling activity in DICE. MORE II is a commercial software package available through Symantec Corporation, runs on the Apple Macintosh, and is therefore readily available and commercially supported. MORE II is a text-based tool, with the associated benefits described above.

MORE II provides a convenient means to edit hierarchies in outline form. A style sheet has been developed for DICE data modeling which provides a convenient template for creating relationship hierarchies such as inheritance trees, assembly trees, and so forth. Features which are supported include collapsing/expanding of hierarchies, and cloning of attributes. Since MORE II is a generic outline editor, any entities can be modeled. However, no special constructs or notations are provided for modeling activities. MORE II is being used both as a simple abstract data modeling tool, and, through the generation utilities, as a data definition language.

PDES/Express
PDES (Product Data Exchange Specification) is an organization whose purpose is to provide a product data specification standard which is powerful enough to allow product descriptions to be represented for exchange between different computer systems, and even between different manufacturers. The intent of PDES is to transfer *information* rather than raw *data*, where information is at a high level with conceptually descriptive information. The benefits of exchanging information rather than data are described in several papers by Peter Wilson[11] [12]. PDES has been accepted as the basis for the future international exchange standard STEP. PDES also has defined a data definition language known as Express.
Express supports the separation of abstract and concrete ideas, since different users have need for different degrees of detail. Express also models the constraints which are to be

imposed upon the things which are modeled and the operations in which the things modeled will participate[13]. Express is therefore an emerging standard for data definition languages. Express was originally intended for abstract data modeling, but current work on Express parsers suggests that Express may become used as a data definition language, as well. Express has mostly emphasized the product model rather than the process model to date. However, recent work has been initiated in creating activity models for DICE using Express, and will be continued in subsequent phases[11].
Express will in the next phase become an alternative to MORE II as input to the DICE generation utilities. The Express language is much richer and more appropriate for information modeling than is the simple MORE II editor described above. These differences were explored in , under Phase II[14], and an Express description of the PPO used in the December demo was created and demonstrated.

Automated Generation from Schema
As described above, MORE II is currently used as the tool by which the formal PPO information model is represented. This model contains entities, parameters, data types, and the relationships between the various pieces of the model including inheritance relationships, as well as assembly and functional relationships. MORE II is capable of exporting an ASCII text file as input to automatic generation utilities.

Class Definitions
One such tool, known as "genSchema",was developed to generate Objective-C source code class definitions, along with data access methods. This process has been called "schema generation". Once compiled, the class definitions can be used to view, manipulate, and update associated data files.
The Objective-C class methods actually create ROSE schema objects and manipulate all data internally as ROSE application objects. This has the benefit of providing the application developer with a higher-level object-oriented interface that is more tailored to the specific object classes and their attributes, while supporting object persistency through ROSE.

File Management
The DICE information system provides a common organization for a heterogeneous collection of files. The directory structure may contain files of any type. Application-specific files are those which would be used even in a non-CE environment. Structured files contain summaries of important parameters, especially those which may be of interest to multiple disciplines, together with references to tool-specific files.

Structured Files

The structured files used by the CE Testbed are those supported by the ROSE database system. The structure definitions are automatically generated from the schema specification by genSchema, as described above. The "update_formats" tool can create format variations variations necessary to support a variety of other host applications, including the "ROSE Format" (ASCII files in pretty-print format which are easily read and understood), "DICE Format" (ASCII files which are not generally human-readable, but which are more compact than ROSE Format), "binary formats" native to ROSE, as well as several additional external formats, including CLOS, FORTRAN *Namelist*, and *Laser* formats. Thus ROSE can serve as an "object server" exchanging data between a variety of

client host applications, and converting its native formats to the client formats on demand. Obviously, differences in client host application capability will dictate the level of symmetry this exchange capability makes available; FORTRAN namelist, for example, only support simple ROSE structures.

Shared PPO Models

An emerging trend in software systems is the sharing of objects between applications. These applications may be running in different processes, and perhaps even on separate machines. Examples of the use of object linking can be seen even in the PC world. The next release of Macintosh system software (System 7) will support linked objects between applications. It will be possible for a text document to share graphics and spreadsheets with the applications which create and edit them. Any modifications to the shared graphics or spreadsheets will be immediately reflected in the text document. By sharing rather than replicating objects, there is no redundancy and no consistently problems.

In DICE, object sharing will enable such things as hot links in which data collected or calculated by one designer can instantly appear on another designer's workstation. This capability has been prototyped and demonstrated using spreadsheets. In particular, MATLAB, running on VMS, has been hot-linked to a process running on an Ultrix machine, which in turn updated values in a spreadsheet running on a Macintosh.

DICE is striving to define a Public Object Protocol (POP) by which objects in different environments, such as Objective C, C++, CLOS, Laser, and so forth, may be shared across processes. In order for objects to be shared, they must consistently follow the common schema, so that there is a correspondence between the remote objects.

Standardized Content

In our prototyping experience, certain elements of the auxiliary models have been determined to be crucial in supporting the CE environment. Some of these have yet to be captured in the formal schema development pipeline, but rather have been mocked up in ad hoc ways to enhance prototypes and demonstrations. Further work is required to formally define and structure the PPO model to contain this auxiliary information. It is conjectured that an enterprise-specific instantiation of this auxiliary information will necessarily become a standard part of any PPO.

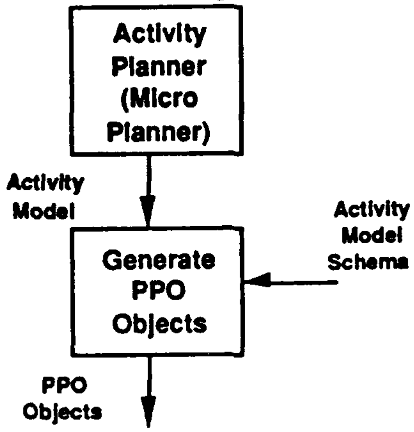
The major categories of this ad hoc auxiliary information that we have used to date include different activity views, events, constraints, tradeoffs, and hypermedia links. Our experience with these models will be described in this section. This discussion provides a flavor for current research directions for the PPO.

Activities

There are many ways of representing activity information. It is assumed that several formats will become standard parts of the PPO model for an enterprise. Several types of activity models have been used in existing DICE prototypes and demonstrations. These include:

- A network of activities which form a PERT diagram.
- A "dynamic" GANTT chart in which posted events dynamically create the chart, filling in entries at each time increment.
- A "dynamic" flow graph in which posted events cause the currently active activities to be indicated.

The PERT representation closely follows the internal structures used by the commercial project planner, Micro Planner™. This representation was modeled using the PPO schema pipeline described above. A utility was then written which would translate Micro Planner™ files into PPO object instances for the class definitions produced by genSchema. This in effect integrated Micro Planner™ into the DICE system as a low-cost activity model editor. The scheme is illustrated i the figure below.



Using a commercial project planner as a PPO editor

Ultimately it is envisioned that the DICE Blackboard will use the activity model as a template for the anticipated tasks being performed by designers. The DBB will compare the actual tasks with those anticipated and determine whether it is appropriate to notify designers of deviations, or to perhaps suggest changes to the original model. The DBB will work in conjunction with activity monitors to provide information related to the current status of the project. The DBB will also add history information to instances of the activity model and store these instances as project histories.

Near-term future work will include integrating the concepts found in Peter Wilson's work[11 14] into the activity model schema.

The "dynamic" models mentioned above run on a Macintosh and generally serve as demonstration aids for tracking the progress of the concurrent scenario. These models, and their associated tools, are not yet integrated into the unified PPO, but instead have their own local activity data. In the future, this integration must be made so that the status monitor tools access the same activity model as does the rest of the DICE system.

Events

One way in which DICE modules have been integrated together for demonstrations is by sharing common Configuration Management and Post utilities. The Configuration Management utilities provide data integration in which the various DICE tools access and share the common PPO. The Post utility provides integration in that all DICE modules issue a common form of status message, or *event*, into the DICE architecture Various modules use these events to coordinate and monitor the activity plan of the concurrent project. These modules include the DICE Blackboard, the Status Monitor, and a utility which displays status information within banners on each workstation.

In the current prototype implementation, different types of events can be posted, and each type has its own list of parameters which it expects to receive. A posted event includes the name of the user or discipline posting the event. In some cases this name can be determined automatically.

In the future, the PPO schema should be extended to include records of the events which are generated during the execution of the CE Testbed, as well as transcripts of Configuration Management transactions and assertions which are made to the DICE Blackboard. In previous demonstrations, the events have generally been posted manually. In the future, wrapped applications should post events automatically.

The transcripts of posted events will provide a design history which can later be used to reconstruct design decisions. Although it is a difficult research area, there is significant interest in attempting to capture design intent as well as low-level design transactions.

Constraints

Constraint management will place additional requirements on the PPO model. Constraint management requires that the DICE system check for violations in design parameter constraints, and perhaps provide additional information such as indicating areas of high sensitivity to design changes, propagating changes through a design in such a way that constraints remain satisfied and so forth. The constraints themselves will need to be represented in the PPO model, whereas the constraints will be checked within the DICE Blackboard. [15]

It is conceivable that constraint management could be integrated with activity management in such a way that the system could suggest activities which need to be performed in order to attempt to satisfy constraints. In other words, the PPO may contain information regarding possible activities which will produce modifications of the particular parameter values which are in violation of some constraint. The result would be a constraint-directed activity management system. The benefit of such a system would be that work would be most concentrated toward improving those parameters which are most critical for constraint satisfaction.

Types of Constraint Systems

The most primitive constraint system would involve algebraic constraints on continuous variables. These algebraic constraints could be equalities or inequalities. In general, these constraints may be called "formulas".

More difficult constraint management involves discrete-valued variables. These often require the equivalent of case statements, rather than formulas.

Even more difficult constraint management involves non-numeric discrete-valued variables. These generally have to do with families or categories of features. For example, in an elevator design project, there are different motor types, different pulley types, and different cable types. Only certain ones can be matched with each other. This requires non-numeric, discrete-valued constraints. This is typical of the general class of "configuration" design problems, in which existing parts must be assembled together.

Constraint graphs can be used either for verifying that constraints are satisfied (which is relatively easy), or for attempting to satisfy all known constraints by modifying values within their allowed ranges. This problem could be solved by exhaustive searching, but clearly this is not an efficient approach. Therefore, more sophisticated heuristics for satisfying constraint graphs are generally used.

The most difficult sort of constraint system, but the one which could potentially yield the greatest benefit, is a system which deals with fuzzy constraints. In this case, a constraint does not have a hard cut-off point at which an unacceptable violation is said to occur. Instead, there is an additional cost involved as the parameter value varies from its ideal. This sort of tradeoff is typical of the type of decisions which actual designers must face.

Propagation within Constraint Graphs
In addition to testing for constraint satisfaction, equality constraints have the further power of acting as "assignment" statements, if it is possible to separate one variable from the others in the formula. This allows constraints to propagate values, since variable values are deterministic, in this case. Propagating values is much less feasible with inequalities, for which an infinite range of values are allowed.
In the propagation of values, an important issue is whether all propagations occur as soon as possible or whether propagation could be suppressed, for performance. In the latter case, the issue of constraint graph consistency is introduced.

Constraints and Activities, Compared
The propagation of values through the constraint graph seems similar to the execution of an activity graph, in which each activity node may provide new parameter values to some model.
However, constraint graphs and activity graphs are not generally combined, or at least have not been in systems developed to date. Constraint graphs do not generally deal with such complications as resources needed in order to enable an activity. Also, inequality constraints do not lend themselves well to the activity graph analogy, since there is no deterministic way of propagating values through inequalities.
The similarities between the propagation aspect of constraint graphs with the data flow found in activity graphs does suggest that it may be conceivable to merge activity and constraint engines into a unified system which performs the functions of both.

Tradeoffs
The goal of tradeoffs is to provide the designer with initial information regarding the cost and benefits of different design options. Ideally, the good design decisions would be made initially, and fewer design iterations would be needed later.
In the case of inequality constraints, defined above, some values within the allowed range may be "better" than others based on some criteria. This criteria could be "superimposed" on the constraints.
Constraint graphs are generally separated by discipline. Inter-disciplinary trade-offs therefore come into play. For example, rounded airfoil corners may be most manufacturable, but square corners are most efficient. The cost of manufacturing and of efficiency could be calculated for various values of roundness, and these could be combined into a trade-off curve. Trade-off curves generally plot some feature within the constraint graph as a function of some other feature, in order to show how variations in one feature affect the other. In the airfoil example, the plots for efficiency and manufacturability may be combined in order to find the value of "roundness" which provides the best trade-off between the two.
For the current testbed scenario, a trade-off is illustrated which involves a stress problem on a blade. A longer shank would decrease stress, but would also increase the weight to an unacceptable value. A curved dovetail would also decrease stress, but would be too expensive to manufacture. The proposed solution is to try a rotated dovetail. The hope is that this will alleviate the stress problem while still satisfying the constraints on weight and manufacturing cost. This level of complexity of typical of real-world design decisions.

Hypermedia Links
The product model itself must capture the form and function of the product so that multiple application programs can access the information which describes the product. The product

model (and, indeed, the entire PPO) should contain a rich set of relationships linking different aspects of the design, so that an application program (and, ultimately, the user) can easily trace between related aspects of the design. These relationships may be organizational (i.e. assembly or functional relationships), temporal (i.e. links to earlier design versions), or keyword (i.e. links to documentation).

For different types of projects, the different types of relationships may take on different levels of importance. For example, in the development of mechanical parts, designers tend to use the assembly hierarchy as the primary type of relationship. In electrical design, functional relationships tend to be more important.

The prototypes which have been developed for the Electronic Design Notebook[16] simulate hypermedia links within the PPO. Besides providing documentation, the EDN will provide crucial information on earlier designs which may impact decisions made on the current design.

Conclusions

The overall approach seems sound:

- The "information pipeline" approach for PPO instantiation works well; the usually tedious job of schema building was done with much less difficulty than usual.
- MORE II is an extremely efficient tool for schema creation, and was used to quickly create a large (1000 class) database; on the other hand, it cannot be used simply to create complex relationships.
- PDES/Express offers great promise as the data definition language for the PPO.
- "genSchema" works well for the automated creation of Objective-C and C++ source code definitions and access methods for ROSE.
- The SCCS-based Configuration Management System built to support the publication and retrieval of any type of file, regardless of its format worked well; its modular integration into the rest of the PPO should permit replacement by commercial packages in those DICE environments where such "legacy code" is already in use.
- Execution time response for the PPO was excellent, particularly when the new "fast binary" format is used.
- No insuperable problems were found with the basic PPO approach, and it seems capable of addressing basic system requirements.

Recommendations

DICE requires a CE information system which supports the heterogeneous information environment, while at the same time providing a high enough degree of integration to enable information sharing and concurrent work.

Simple and effective tools must be created for the PPO in the short term which support these goals. The major foci must be on developing schema to support the most important design parameters, developing generator utilities which facilitate the translation from schema to implementation, supporting data access in a variety of languages and formats, and providing utilities for publication and notification. Prototypes have been developed in these areas which will continue to evolve within the DICE CE Testbed.

Major research areas for longer term work will include the integration of constraints, tradeoffs, and dependencies within the PPO. Also, for performance, the PPO will need to

be made distributed, and this will lead to the requirement of a data dictionary function for locating necessary data within the distributed information system.

Specifically, we should continue and extend the basic PPO approach:

- Capture the "information pipeline" in an integrated hypertext document so that all definitions are traceable back to the original materials.
- Since the schemas which define these structures evolve iteratively, the CE environment inherently requires a data system which supports flexible, dynamically changing schema and dynamic migration of instance structures between schema in order to support these iterative changes.
- Put together a context-sensitive Emacs/Express textual editor to generate Express schema definitions quickly and efficiently.
- Continue extending PPO schema to include records of the events which are generated during the execution of the CE Testbed, as well as transcripts of Configuration Management transactions and assertions which are made to the DICE Blackboard.
- Prototype examples of key features of the PPO "auxiliary" models including constraints and tradeoffs.
- Support object exchange for other languages in addition to Objective-C, such as FORTRAN, C++, Laser, and CLOS.

---

[11] Wilson, Peter R. "Information and/or Data"?, IEEE Computer Graphics and Applications, June 1987

[12] Wilson, Peter R. "Information Modeling", IEEE Computer Graphics and Applications, June 1987

[13] "Information Modeling Language Express", ISO TC184/SC4/WG1 N287, October 1988

[14] Wilson, Peter R. "A Short History of CAD Data Transfer Standards", IEEE Computer Graphics and Applications, June 1987

[15] Londono, F.; Cleetus, K.J.; Reddy, Y.V., "A Blackboard Scheme for Cooperative Problem Solving by Human Experts", Proceedings of the MIT-JSME Workshop on Cooperative Product Development, Nov 20-21, 1989.

[16] Uejio, W.H., "Electronic Design Notebook for the DARPA Initiave in Concurrent Engineering", Proceedings of the 2nd Annual Symposium on Concurrent Engineering, In Press (1990)

Task 3.3.3.5 Information Manager Node

Objectives

The purpose of the DICE Information Manager (IM) is to give an overview of what is happening within the DICE environment, and to provide tools to help determine what could be and should be done within the environment. The IM therefore provides activity management tools for project planning and tracking, as well as tools for running "what-if" scenarios, which help determine the best paths to pursue during design evolution. In short, the Information Manager is a tool which provides a "road map" into the concurrent project. The information provided by the IM is available to all project participants. The tools for project planning may be available only to the project manager.

The IM also must provide tools for browsing PPO schema and data and management tools for generating PPO schema.

The IM activity management tools must track the various versions which are under development within the concurrent engineering environment. It would be difficult to keep track of which people or groups are performing which tasks, using which versions, without project tracking tools to facilitate this. The project plans and status which are generated and tracked by the activity manager will be stored in the PPO.

Tools to run what-if tests against analytical models provide an invaluable tool for choosing the correct design paths which should be pursued. Analytical models provide a first-order approximation of the characteristics that a given design will have. Running analytical models against a range of possible parameter values suggests the parameter values which hold the most promise. By integrating the what-if tools with the PPO, first-order evaluations can be determined against the archived library of existing designs.

Finally, the IM should demonstrate an *electronic "design notebook"*; the EDN has the following goals:

- Support hypermedia documentation
- Capture design intent
- Support cut and paste from applications
- Preserve linkages back to the PPO for pasted objects
- Allow users to re-launch "recorded design activities"; the EDN should function as a window on the design environment.

Approach

A rapid prototype of the Information Manager was implemented on a Macintosh SE:

- WingZ was used as an application prototyping environment.
- Communication links to UNIX workstations were established by means of shared directories between the Macintosh and UNIX. This is possible through TOPS (for Sun workstations) or through AppleShare (for Ultrix workstations).
- Prototype EDN as a WingZ application.
- Mathematica was used as an prototype EDN.

- Leverage spreadsheets as a convenient mechanism for organizing data, allowing functions and dependencies between data, and for providing a user interface over tables of data. WingZ was chosen because it provides this spreadsheet capability, along with graphics, Hypercard-like links, and programmable event handlers, all of which can be controlled and programmed using the Hyperscript programming language.
- Integrate a commercial project planner "MicroPlanneer Plus" with the DICE envoronment for use as the system project planner.

Technical Results

Object-Oriented Environment

In order to facilitate the development of the IM in WingZ, a WingZ Object Manager was implemented. The purpose was to provide an programming environment in which the WingZ Hyperscript code could be well-structured and readable.

Trade-off Spreadsheet using Parametrized Models

The IM prototype showed several other tools which are useful in determining future design paths. One such tool is a set of curves which show how the sensitivity of stress and vibration characteristics change with the geometric properties of a turbine blade.

"What-If" / "Hot" Spreadsheet

Another tool used the data from the trade-off curves shown above. This is a "what-if" spreadsheet which calculates values of stress and vibration sensitivities given a table of geometric dimensions. This spreadsheet is also "hot" in that the input table can be modified based on geometric parameters returned by a wrapped method code to the XS spreadsheet, which runs on an Ultrix workstation. Updates to the XS spreadsheet should therefore vary the input parameters of interest in the IM "what-if" spreadsheet. These input parameters in turn map through the trade-off curves to arrive at values for stress and vibrational sensitivity.

EDN

EDN Implementation

WingZ

In the Phase II effort, the spreadsheet package *WingZ* was used to implement a "Pad of Paper" level EDN on a Macintosh. With the WingZ programming language, we were able to read files and launch other applications. The data from these applications were then brought into the spreadsheet with filters and saved in local files. The data was easily displayed by WingZ. A local database was maintained to aid in maintaining the links and link reconstruction when links were added or deleted. In summary, the WingZ implementation for the "pad of paper" worked very well.

Mathematica

In the Phase II effort, the symbolic mathematics tool *Mathematica* was used to implement a "Laboratory Notebook" level EDN on a Macintosh. Within the Mathematica programming language, we were able to read files from the PPO, launch other applications and utilize external procedures. The data from these applications was subsequently mapped into the Mathematica notebook, and displayed. In summary, the Mathematica

implementation for the "Laboratory Notebook" worked very well. The interface is unusal, but easy-to-learn, and provides the user with a powerful tool with transparent access to the other DICE services.

Integration

The communication between the Information Manager running on the Macintosh using WingZ is done using TOPS/AppleShare to create a shared directory between the Macintosh and the Ultrix file server. Since WingZ supports user-supplied "on-idle" event handlers, it was possible to create a WingZ Hyperscript function which looks for files in the shared directory whenever the Information Manager is not otherwise engaged.

In the demo, a UNIX script containing the status update messages for the demo scenario was stepped through from a remote terminal in synchronization with the demo. Ultimately, of course, the concurrent designers and the wrapped applications should themselves send the status updates.

The inverse communication path has also been implemented. WingZ can write a text file to the shared directory. A different Ultrix server then executes this text file as a UNIX script.

Conclusions

The Information Manager Prototype developed under Phase II illustrated many of the capabilities that DICE will provide later in the program, and was well received by users. Many of the weaknesses revealed in the Phase I implementation were eliminated,

- The commercial project planner, Micro Planner™ was integrated into the system as a low-cost, easy-to-use activity model editor. This will allow the status monitor tools access the same activity model as does the rest of the DICE system.
- Framemaker and Mathematica were evaluated as alternative EDN vehicles, and proved be well suited to the requirements for the formal EDN and the "Engineer's Notebook" EDN.

Recommendations

The basic ideas are sound and should be pursued through more detailed prototypes.

- Link the IM more directly to the PPO in ensuing versions. Private databases stored in spreadsheet memory in the current version should be moved to the PPO and versioned controlled.
- Implement the capablity for issuing task assignments, autonomous execution of well-behaved, deterministic application processes, etc.
- More robust communication mechanisms are needed between communicating modules.
- Integrate a Calendar System in the EDN which would lend itself to a temporal search methodology.
- Explore methods for automatic translatiion between the different EDN levels
- Test the user acceptance of the EDN with real users.
- Implement the ability to capture the journals of other applications in the EDN under user control.

Task 3.3.3.6 Integrated Operating System Thread

Objectives
The major objectives of this task were to pull together the separate components and "nodes" of the DICE CE architecture into an integrated system, prior to moving it to Morgantown WV for final integration with the CERC developed architecture modules prior to the Phase I demonstration.

Approach
Since the internal development environment at CRD is Ultrix-based, the bulk of early development was carried out in that environment. Subsequently, CRD developed tools were ported to the actual target platforms used in the demonstrations.

Technical Results
Integration of a functional subset of the DICE CE architecture was achieved in the CRD Ultrix environment. Other components, principally the Information Manager Node and the Manufacturing Workstation, were integrated later. The Manufacturing Workstation running under VMS was also coupled, but still rather loosely in the Phase II demonstration. Since the DICE Information Manager node runs on a Macintosh, that integration was carried out in the Macintosh environment, and was also loosely coupled.

Conclusions
The integration went smoothly in a uniform, common single operating system environment, however when the software was ported to other platforms inconsistencies were discovered which were difficult and time consuming to locate and resolve.

The fact that CRD did not have local access to some of the hardware platforms used at CERC posed porting and integration problems not encountered until final integration at CERC.

Recommendations
Put in place at CRD and at CERC similar hardware and software, so that software porting problems can be identified and corrected earlier in the development cycle.

DICE PROGRAM

Task 4.3.3.7 PaLS **NCSU**

1 Objectives

Traditional methodologies of product development are based on a sequential flow from specifications to a detailed design which is then manufactured, tested, and delivered to the customer. In reality this almost never happens. For example, a product specification may progress to the production phase before it is determined that it is not manufacturable, or that manufacturing costs for the product as designed are prohibitive. In the worst case, the project goes all the way back to the R&D phase. Such long feedback paths result in long product development times.

An alternative approach to the traditional methods described above is to take a more fine grained view of the operations and interactions required to go from product concept to delivery. The process may be represented as a directed graph, where the nodes of the graph represent primitive operations such as *evaluate the thermal properties of material Z*, or *design widget A*, or *inspect assembly B*. The edges coming into a node then represent the information required to perform the operation, for example the *functional specification for assembly B*, and so on. This approach tends to elucidate the explicit dependencies among all of the operations required in the product development cycle. The problem then becomes one of mapping the nodes of the graph onto the available resources (people and machines) and scheduling the operations assigned to each resource so as to achieve maximum concurrency.

By so doing, the product development cycle will be minimized. This approach to product development is one aspect of the current DARPA Initiative in Concurrent Engineering (**DICE**).

When maximizing the concurrency of the design process within given cost constraints, the following questions must be answered (among others):

- What types and amounts of resources will be needed to complete the design on time?
- How will the workers be organized to minimize communication delays and errors?
- to whom will the tasks be assigned?
- In what order will the tasks be performed?

The **DICE** scheduling problem is difficult to solve due to its combinatorial nature - i.e., the quality of a solution is affected by a large number (possibly millions) of interacting decisions. Expressed mathematically, we must find a vector $\mathbf{s} = \{s_1, s_2, \cdots, s_N\}$ which minimizes some function of interest, $H(\mathbf{s})$, that depends on $\mathbf{s}$ in some complex, non-linear way. Compounding the difficulty, the problem is *discrete* in that the decisions may assume only one of a limited number of values (e.g., $s_i \in \{0, 1\}, \forall i$). Typically, the decisions are made based upon a combination of prior experience and guesswork. Unfortunately, experience can bias a schedule away from an original and advantageous configuration, and a single poor guess can distort an entire system due to interactions between decisions.

2 Approach

At North Carolina State University, we are examining the use of simulated annealing and neural networks for solving optimization problems without bias. Simulated annealing is a gradient descent technique incorporating a random process which allows the escape from local minima such that the globally optimum solution can be found. Neural networks contain many simple processing elements which are highly interconnected such that they can rapidly solve large problems in a cooperative manner. We have incorporated the randomness of simulated annealing into neural networks to create the Mean Field Annealing (MFA) algorithm. The controlled randomness improves the solutions found by the neural network, while the cooperative and continuous nature of the network increases the speed and parallelism of the annealing process. Thus, MFA can rapidly find near-optimal solutions to a wide variety of problems.

MFA solves scheduling problems by manipulating the following variables

$$s_{ijk} = \begin{cases} 1 & \text{if task } i \text{ is executed on resource } j \text{ at time } k \\ 0 & \text{otherwise} \end{cases}$$

so as to arrive at a near-optimal schedule. The variables are updated according to a normalized Boltzmann distribution as follows:

$$s_{ijk} = \frac{\exp(-\Phi_{ijk}/T)}{\sum_{l,m} \exp(-\Phi_{ilm}/T)}$$

where Φ_{ijk} is merely the cost incurred by executing task i on resource j at time k (i.e. $s_{ijk} = 1$). Lowering the control parameter T (often called the *temperature*) forces each task to converge to the resource and time slot having the lowest overall cost. This cost is of course dependent upon many of the other task assignments, thus the tasks cooperate and compete for desirable resources and time slots and eventually reach a near-optimal global solution.

3 Technical Results

We have modified and repackaged **PaLS** to form **MFTP** — the Mean Field Task Planner which optimizes task schedules in the DICE environment. **MFTP** is a tool to be used by the DICE Project Lead (PL) to plan and evaluate the distribution and scheduling of task assignments over the available organizational resources. The PL interfaces to **MFTP** via **XS** — an X11-based spreadsheet. We also created a software tool, **GanttView**, to allow the optimized schedule derived by **MFTP** to be graphically displayed as a Gantt chart in a workstation window.

In May we successfully demonstrated the function of **MFTP** on a small problem as part of the "executable mockup" at the dedication of CERC in Morgantown, West Virginia. In solving the demonstration problem, we found that the execution time is polynomial in the product of the size of the task graph, the number of available organizational resources, and the length of the scheduling time horizon. This creates a significant problem, particularly when we attempt to scale up to real problems of interest in the context of DICE. Therefore, a new version of **MFTP** was created which progressively halves the scheduling horizon and determines in which half a given task will reside. For a horizon of T time units, the new algorithm requires $\log_2(T)$ iterations to achieve the same precision as the original version of **MFTP**. However, the execution time of each iteration is now a polynomial of only the size of the task graph and the number of available organizational resources. This led to an overall improvement which allows **MFTP** to be used to interactively on more realistic problems.

The divide-and-conquer approach to scheduling tasks over a time horizon of length T reduced the execution time of **MFTP** by a factor of $\log_2(T)/T^2$. By applying clustering techniques to the tasks and resources, we hoped to gain similar reductions in problem dimensionality

with a resulting increase in execution speed. The decrease in dimensionality is achieved by clustering sequential or near-sequential tasks with similar resource usage characteristics into *super-tasks* (STs). These STs are then assigned for execution on a given resource at a specific initiation time. Such clustering can be done using Kohonen neural networks, competitive learning networks, or standard hierarchical clustering techniques. We have also developed two new clustering techniques based on the MFA and Kernighan-Lin heuristics, respectively, as a result of this work.

A significant portion of our effort has been expended on enforcing hard constraints in the solutions generated by MFTP. MFTP can only enforce soft constraints since it uses a penalty function with an associated Lagrange multiplier for each constraint. We have developed the *technique of logical consequences* (TLC) which guarantees the satisfaction of precedence constraints in the scheduling problem *without requiring* a *costly* Lagrangean multiplier adjustment phase in the MFA algorithm. Recent investigations into the use of TLC on simpler benchmark optimization problems have removed some of the restrictions on the technique. These investigations have also shown that TLC can fail to enforce certain non-precedence constraints when the states of the cooperative units in the MFTP are highly correlated.

To gain more speed, the optimization methods used in MFTP were examined with the intent of reducing their computational complexity. Resulting gains of 20% – 30% in computational speed were deemed insufficient when compared to the improvements obtainable by using hierarchy. Increasing the speed of MFTP by using vector and parallel processing as also investigated on our Ardent superminicomputers. As with the code optimization, the effort required to increase the speed of MFTP by even a factor of five was found to be excessive. However, increasing the speed of the simpler clustering algorithms is more easily achieved.

This can have a multiplicative effect on the speed of the entire scheduler since clustering reduces the problem complexity and greatly increases the speed of convergence of MFTP.

4 Conclusions

From our experience with using the MFTP on the DICE scheduling problem, we have found:

1. MFA can effectively optimize a wide range of forms of cost functions expressed as a function of simple binary decisions. The large number of decision variables slows the convergence of MFA. Applying clustering techniques to hierarchically structure the optimization problem can bring about dramatic increases in speed.

2. Constraints on the problem solutions can be expressed in the cost function using Lagrangean multipliers, but the constraints can be violated as MFA converges. The use of TLC can usually prevent these violations from occurring and reduces the number of Lagrangean multipliers needed.

3. Vector and parallel processing appear effective at increasing the speed of the clustering algorithms used to hierarchically structure the scheduling problem.

5 Recommendations

Based upon the previous conclusions, the following areas of research are recommended:

1. The clustering techniques for extracting the hierarchy of the scheduling problem must be improved.

2. Use of parallelism to speed the clustering must be further investigated.

3. Improvements to **TLC** must be made to further reduce the possibility of convergence to an infeasible schedule.

6 Publications

"Encoding Logical Constraints into Neural Network Cost Functions", D. Thomae and D. E. Van den Bout, to appear in the *Proceedings of the IEEE International Conference on Neural Networks*, 1990.

7 Hardware/Software

	Hardware Purchased	Qty
1.	None	0

Software	Description
MFTP	the Mean Field Task Planner
GanttView	an X-11 based graphics program for viewing the schedule output by **MFTP**.

DICE Program

Task 4.3.3.10 Knowledge Server Bell Atlantic Knowledge Systems, Inc.

Objectives:

The Knowledge Server is a component of the DICE architecture that enables engineers to access and view information stored in the enterprise database. The objective of this task was to enable the Knowledge Server to access external information and to improve the user interface and data access operations that were provided to the user.

Approach:

The Knowledge Server that was implemented during Phase I of the Knowledge Server was primarily able to operate on self-contained information. A multiple process architecture was developed consisting of a primary Server process that interacted with a number of user interface processes. Each user had a separate user interface process which channeled commands from the user to the Server and results from the Server back to the user. Users were able to view information using print or draw commands, or search through the knowledge-base via the query command.

In order to achieve the objectives of the Knowledge Server, the following changes were made to the Knowledge Server software:

- its data access operations was modified to enable interaction with external information;
- a communications protocol to exchange information between processes was defined and established and
- its user interface was improved and functionality for various interactive operations developed.

Technical Results:

In order to facilitate exchange of information between the Knowledge Server and other processes, a communications protocol was defined and developed. A process, simulating the role of the DICE Black Board, was created so that this concept could be tested. During demonstrations, users had access to information within the Knowledge Server's own database or controlled by the simulated DBB.

The data access operations of the Knowledge Server were modified to operate upon external information. This was successfully demonstrated for discrete information (both print and display commands).

The SQL module of the Knowledge Server's User Interface was upgraded and as a result it:

- Enabled queries to be formed with correlated subqueries;
- Implemented an extension to standard SQL by means of the ASSIGN keyword by which the results of query may be stored for use in future queries;
- Expanded the use of expressions and set operations in queries;
- Developed a generic SQL-module that can be embedded into other processes. With this module and the distributed SQL interface it will be possible to process queries on distributed data sources and
- Provided a SQL keyword dictionary for the user interface to enable queries to be formed easily and correctly.

The iconic user interface of the Knowledge Server underwent considerable upgrade:

- The operation of the display command was enhanced by incorporating functionality for hyper-card-like context-dependent hierarchical browsing operations. Two-dimension scrollbars allow examining oversize images;
- The print command operation was upgraded by improving the output format of objects. Units are now displayed for each attribute of an object,

and the use of two-dimension scrollbars enable browsing of large objects. The ``what-if" analysis operation was enhanced by utilizing external routines for complex computations;

- Context-sensitive help messages were included to enable ease of operation by end-users and

- The PPO classes were defined in LASER and knowledge-bases were created to enable demonstration of the Knowledge Server's capabilities in December 1989 and February, 1990 demonstrations.

Conclusions:

The functionality of the Knowledge Server was expanded to operate upon external information. It is able to print and display objects that are stored in external knowledge-bases or individual files containing object information in LASER format or it can exchange information with other processes.

However, since the DICE information is to be stored in formats other than LASER, it is imperative that the Knowledge Server be able to operate upon many information repository formats. This can be achieved by means of defining/developing the means to translate information from one format to the other. Such a functionality will be required for any program that accesses information that is stored in the DICE environment.

An attempt was made to establish a DICE-wide uniform user interface specification --- namely TAE+. However, the version of TAE+ software that was available during Phase-2 was not capable of generating dynamically-defined windows (a necessity with the Knowledge Server which requires a number of windows of various sizes).

A significant shortcoming of the DICE project, at this stage, is the absence of an enterprise data model definition. Such a data model is imperative if programs are to be developed that operate upon this data. The absence of such a model is a significant roadblock in the path to construct and integrate architectural components.

Demonstration quality data which fits the data model definition needs to be made available so that programs can access such realistic information and can generate realistic demonstrations.

Recommendations:

The Knowledge Server should provide an inter-process communications interface via which its services should be provided to users as well as programs. It needs to browse through the contents of the PPO, displaying object contents, object hierarchies and documents associated with those objects. It should be well-integrated with the rest of the DICE architecture utilizing the services of the PPO, the LCM and other such systems. It should adhere to DICE data exchange standards which are to be established during Phase 3. It should enhance its SQL interface and attempt to interact with external relational databases. Another area of improvement should focus on providing means for displaying graphical entities.

Publications:

Wong, Dennis H., Pedro Oscar Cubillos, Ravi S. Raman, Michael Bender, David Dymm. "The Knowledge Server Module of the DICE System: Phase II." In Proceedings of the Second National Symposium on Concurrent Engineering. February, 7-9, 1990. Morgantown West Virginia.

Hardware:

No computer hardware was purchased or developed in this task.
No computer software was purchased in this task.
Computer software for the User Interface and Server processes of the Knowledge Server was developed.

Task 4.3.5.1 Integration at CERC — GE-CRD

Objectives

The major objectives for this activity revolved around the preparation for and execution of concurrent demonstrations at CERC. Specifically, the task objectives were to:

- Reach consensus with the CERC (and other DICE program contributors) on CE architecture goals, directions, and demo-specific implementations
- Integrate GE-developed architecture modules and GE Aircraft Engine supplied application methods, tools and advisors with CERC architecture methods and CERC-developed application methods, tools and advisors into a functional system.
- Carry out a concurrent demonstration of the DICE-developed CE system, illustrating its architecture, methods, tools and advisors for DARPA and other selected reviewers.

Approach

A multi-phase approach was pursued in Phase II, consisting of:

- Concurrent scenario finalization/demo planning
- System integration
- "Demobook" preparation
- Concurrent Demonstration

Technical Results

Team building, scenario finalization and demo planning were important and successful activities, but since they did not *directly* produce tangible results not covered elsewhere in this document, they have not been reported on. Specifically, listed below are the concrete results of activities associated with this task.

"DemoBook" Preparation

A "DemoBook" describing the December demo was prepared in December of 1989.[17] The "DemoBook" attempted to convey the DICE vision, the scenario the DICE team planned to follow in the demonstrations, as well as thunb-nail descriptions of the architecture modules demonstrated under the demonstration.

System Integration

Integration of architecture and application methods software produced by GE and CERC/WVU was initiated several months prior to the December 1989 Phase II DICE demonstration. In spite of the extremely tight initial integration schedule, significantly more complete integration between GE-developed and CERC-developed software was demonstrated in the Phase II December demo then was apparent during the Phase I demo.

Concurrent Demonstrations

The objective sof the concurrent demonstrations were (1) to show some of the different ways in which product development can be made concurrent; (2) to work through transactions which show how the information systems framework expedites concurrent product development, and; (3) to serve as a starting point for organizations developing concurrent product development systems. However, the interactions among roles, transactions, applications tools, and systems services are very complex in real concurrent

product development activities. As a result, concurrency is hard to understand in the usual linear viewgraph presentations, in conventional written specifications, or in sequential computer demonstrations. The major challenge in a concurrent demonstration is presenting the multitude of parallel activities and underlying transactions without confusing the participants.

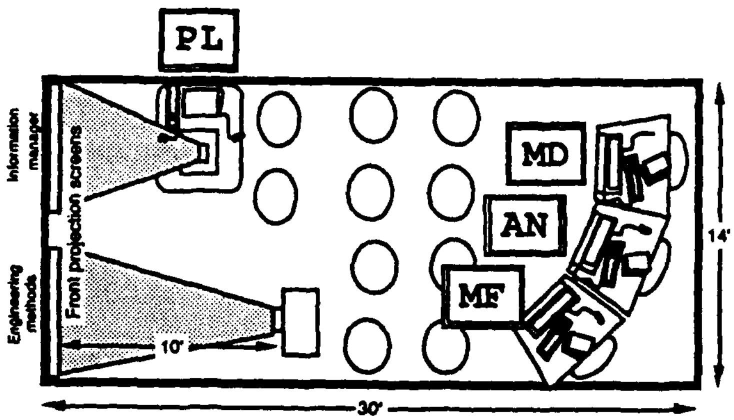
The focus of the Phase II demonstration was a pair of one-hour program tailored for senior technologists and management, and adressing both the mechanical structural domain as well as the electronics domain. Major features of the demonstrations included:

Methods: While some engineering analyses had to be precomputed due to the time constraints of a compressed-time demonstration, two or three iterations of a valid product development process were shown.

Services: At key points, some of the details of the underlying system services were illustrated by examining underlying schema, files, data flows, and data structures.

Products: The Phase II demonstration simulated the production of a part derived from design parameters defined during the demonstration.

In the present one-hour concurrent demonstration, the four members of the demonstration team (e.g., MD, AN, MF, and PL for structures) work through a compressed-time execution of the concurrent scenario (with some variability) using four workstations and their supporting servers. In the physical layout shown in the figure below, the project lead (PL) workstation and its activity monitor are always projected on one screen in order to help the reviewers track the scenario as it unfolds. Either the activity diagram or the concurrent scenario, (they are both described in the section on Task 3.3.3.5.2), were displayed on the activity monitor.



A physical layout for structures demonstration at CERC

Conclusions

- The CE prototype demonstrated in Phase II proved to be a valuable vehicle for developing an understanding of the concurrent engineering process, for testing the DICE information systems framework, and for communicating between the DICE vision.

- The focus on the dvelopment and demonstration of the the CE Testbed allowed the DICE teams to distinguish cleanly between "future" architecture, methods ideas and extensions, and those actually being developed *and demonstrated* in the current phase. As a direct result:

- Consensus was finally reached between the geographically distributed developers on the major elements of the DICE architecture.

- A "demonstration book" describing the modules which make up the current demonstration, along with a concurrent desription of the actual scenario enmployed during the demonstration.[17]

Recommendations

- Revise the demonstration planning schedule to reflect more realistic estimates of the time needed to achieve full integration.
- Future activities should involve expanding the pilot processes, extending the demonstrations, and enabling additional flexibility in design choices.

---

[17] "Concurrent Demonstrations of Structures and Electronics Testbeds", December 15, 1989, Morgantown, WV.

Task 4.3.5.2 Integration at GEAE

Objectives

The major objective in this task for Phase I was to provide realistic problem focus, requirements and specifications for the CE architecture development and subsequent demonstrations.

Approach

The objectives were pursued through a sequence of activities which provided the framework for later developments. The activities were:

- Acquire a representative pilot or model (subset) problem, broad enough to encompass the key CE issues, and small enough to be credibly addressable with the resources at hand.
- Collect descriptions of the product/process activities used in the model problem
- Capture these major transactions for the model problem in the form of an as-is storyboard.
- Evolve a to-be storyboard, incorporating concurrency and addressing future user needs.
- Create a concurrent scenario.
- Identify appropriate application methods tools and advisors to support the scenario.
- Integrate GEAE-supplied tools and methods in a GEAE CE testbed, incorporating the DICE architecture elements.
- Utilize the CE Testbed on a real problem to demonstrate the effectiveness and business benefits derived from the DICE architecture.

Technical Results

The results of these technical efforts are organized by activity, and are listed below.

Model Problem Specification

The key requirement in identifying a suitable model problem is that the activities in the specific model problem be representative of product development in the organization as a whole. If this requirement is met, a system designed for the pilot activities will scale up smoothly into a solution for the whole organization

For an aircraft design example, the aircraft itself is too complex; a major subsystem such as an aircraft engine is too complex; even engine subsystems such as the fan, compressor, combustor, and turbines (HPT and LPT) are too complex. A good choice for a model part is a single turbine blade. Four alpha site candidates were identified within GEAE spanning the life-cycle for aircraft engines. They are:

- Conceptual design
- Detailed design
- Process development
- Logistics

Product/Process Descriptions

Once candidate parts were selected, descriptions of the activities involved in developing, producing, and supporting the part were captured and documented. Careful selection and analysis of these activities was more important to the ultimate success than the choice of the part itself. Because the full collection of activities is too complex to be captured in detail, informal data flow and structure diagrams were most helpful in getting started. Developing these diagrams was surprisingly difficult because the process understanding is usually spread over a large number of people within multiple groups and extensive formal documentation of the product development process is usually not maintained. Finally, the dataflow diagram documentation tool "Design" was used to capture the process descriptions more formally.

Concurrent Storyboards

After the target product development activity was defined and an initial structured analysis completed, the next step was the construction of a concurrent storyboard. The objective in this activity was to define the major transactions in the product development activity so that a smaller and more detailed concurrent scenario could be selected and documented in detail. Most useful are an *as-is* scenario for understanding the current design process (which tends to be a long strung-out sequence of steps) and the concurrent *to-be* scenario for the CE testbed (which usually involves three to five parallel tracks).

Concurrent Scenario

Once the target product development activity was defined and the initial structured analysis (data flows, data structure, and concurrent storyboard) completed, a small subset of the sheets was selected and the transactions worked out in detail. Much like the storyboard, each row of the concurrent scenario is a left-to-right time sequence of boxes representing the tasks for the particular role. Each column of the array represents a particular time interval. Again dependencies, major dataflows, and significant events can be indicated by lines, markers, and notes superimposed on the array. Reviews of the storyboard will usually generate changes in roles and focus from the storyboards.

The concurrent scenario is divided into several phases (usually three to five) representing initialization, a few design iterations, and termination. Each phase contains perhaps three to five tasks. In DICE, the concurrent scenario is captured in two different forms. The summary form of the concurrent scenario diagram (shown above) is displayed on the activity monitor during demonstrations and used as a management planning aid while the DICE system runs. Each of the blocks includes the task identifier (e.g., AN.III.3), an identifier indicating the activity being performed (e.g., EvalFEA), and a list of events posted by that task. In the abbreviated form the focus is on transactions to and from the Product-Process-Organization (PPO) model (which is described in more detail later). The events are described in a simple structured English. Typical events include:

Rel	Release	Release file into the PPO; notify interested parties, and distribute.
Chk	Check	Check out files from the PPO in order to modify the design
Use	Use	Check out files from the PPO; subscribe to change notices
Alrt	Alert	Post an alert concerning and event or problem

Conclusions

The sequence of activities in this task turned out to be an excellent way to address the objectives. Storyboards and concurrent scenarios are intuitive, easily understood, and elicited feedback from a much broader spectrum of end users than we would have obtained had we adopted a more formal methodology as the communications medium.

Recommendations

Continue using the approach outlined above in the next program phase.

DICE Program

Task 4.3.7 Implementation Language **Bell Atlantic Knowledge Systems, Inc.**

Objectives:

A concurrent engineering environment requires not only an architecture that is conducive for cooperative work but also tools and services that embody this concept. Development of such software requires non-traditional software development languages and methodologies which assist in gaining those goals. Object-oriented Languages are gaining popularity due to the provisions of data encapsulation, data-abstraction, message-passing and inheritance that are inherent in these languages. Frame-style knowledge representation languages employed in AI environments provide many of these features as well as multiple programming paradigms and improved data access controls. Software facilities that are important assets in a concurrent engineering environment were identified and developed into a software library, making them beneficial to other application developers in the DICE environment.

Approach:

An examination of the software facilities that may be beneficial to DICE software developers revealed the following needs:

- The DICE effort has embraced the concept of object-oriented programming as beneficial in many software applications. Information is to be stored in object-oriented databases where large complex application objects will contain information about engineering products being designed. In order for data to be transferred between processes, these complex objects (which may be composed of hierarchies of classes) may need to be collapsed into conglomerate objects that may be transported easily;

- Traditional object-oriented languages permit compile-time specification of an object class. However, instantiation of those object classes do not occur until run-time. Creation of these instances of objects can result in delays (especially if it involves a large number of objects). By means of a preprocessor, it will be possible to specify the creation of the object classes and their instances at compile-time, resulting in improved run-time performance and

- There is a need for construction of object classes that enable sharing of information between processes --- for example by leveraging the DICE architecture's LCM facilities. It is also necessary to enable programs to concurrently operate upon parallel knowledge-bases. Finally, since engineering software typically employ iterative operations, support for dynamic array definition, and methods to operate upon them is needed.

Technical Results:

The Preprocessor was implemented as a utility that operated upon one or more files containing object class and instance definitions written in a stylized (easy-to-read) format. The output of the preprocessor consisted of corresponding C language files containing datastructure definitions, data initializations, and some book-keeping information. These files were compiled along with the application's source files, and the resulting executable can now be invoked with the objects in-memory at the start of the program. A small amount of book-keeping operations were completed at the start of the program, but it was negligible compared to run-time object/instance creation. A ratio of about 16:1 in cpu time for preprocessor objects vs run-time created objects was recorded for a collection of 937 objects.
A communications class object was defined that handled IPC communications. With the aid of this object, it became easy to develop programs that employed inter-process communications to share information and services. This class was employed to demonstrate concurrent access to shared knowledge-bases. This example showed how knowledge-bases can be effectively partitioned, reducing their size and resulting in improved data consistency and also improved performance.

A facility for dynamic array definition was implemented. This facility enabled multi-dimensional, mixed data-type arrays to be dynamically created, and enabled the appropriate data access operations (initialize, set, and get) to be performed on the arrays. Traditional engineering software that exploited iterative operations upon arrays can now be performed upon frame style objects.

Facilities were developed to improve object-oriented operations. Such facilities enabled easy definition of object classes, the creation of complex composite objects, improved messaging features that enabled pseudo function overloading, function invocation based upon keyword and parameter type and default parameter values. Such features enabled object classes to be easily implemented, resulting in a code reduction of over 50% in some cases compared to classes created with previously available facilities.

With the aid of these OOP facilities, classes were developed for the LCM communications library. Interprocess communications can now be achieved very easily. Another class developed was a matrix class which exploited the array data-type that was defined earlier and provided creation, addition, subtraction and multiplication operations on these matrices. In addition, classes were defined to assist in ErrorHandling and for Laser programming.

Conclusions:

A software library which contains the various software modules --- preprocessor, object classes, support facilities, etc. -- was developed and installed at CERC. The preprocessor contains a small amount of code which is system-specific. This is necessary because of the manner in which it allocates code for proprietary data structures, and it manipulates such structures. However, despite creating the class structures at compile-time, it does not prevent such structures from being modified at run-time (a facility not provided in any other system).

The object-oriented support facilities are another significant development of the DIL project. They provide superior facilities for creating new classes, especially within the DICE framework and will enable a host of new classes to be easily defined and developed. The new classes were developed with ease and the reduction of code, compared to the previous OOD facilities, was significant.

A seminar to impart information about the DIL library and the facilities offered for software developers was conducted at CERC at the end of Phase2.

Recommendations:

Task members recommend the following activities:

- Develop more classes that facilitate interaction with and exploit the services of the DICE architecture;
- Develop object classes that provide data definition and associated methods for entities such as bags, sets and collections;
- Develop object classes for user-interface development that can be easily embedded into application programs and
- Define and develop methodologies for exchanging data with other programs by interfaces to data exchange protocols.

Publications:

The results of the research and development performed during this project have not yet been published. However, an abstract of a paper on this work is being reviewed by the DICE program administrators prior to submission to a technical conference.

Hardware:

No computer hardware was purchased or developed in this task.
No computer software was purchased in this task.
A software library, which contains various tools and utilities developed during this task and which can be invoked by software developers of concurrent engineering programs, was created.

DICE Program

Task 4.3.9 Soft Prototyping **Bell Atlantic Knowledge Systems, Inc.**

Objectives:

The Soft Prototyping Facility is a utility to assist engineers in evaluating their designs from the viewpoint of manufacturability and in anticipating/examining the consequences of design decisions upon the product.

Approach:

The design verification process was broken down into two steps -- a preliminary check for manufacturability and producibility simulation. The manufacturability issues can be considered via an interactive constraint checking tool that has information about the product design, the manufacturing process and equipment data. By determining constraint violations, the system can identify causal relationships between product design parameters and manufacturing issues. To simplify the process, product design information was broken down into hierarchies of features. Primitive features had information indicating manufacturing processes by which they could be fabricated. Each manufacturing process had constraints under which it could be employed and information about machines wherein they could be performed. In addition, machines had operational characteristics in the form of facts and constraints.
For more detailed analysis, simulation models that can examine complicated interactions of the product manufacturing process that may be affected by or affect the design process needed to be generated. Knowledge-based simulation systems enabled in-depth analysis of the components of the model.

Technical Results:

A generic constraint-evaluation system which operated upon complete or partial product design was implemented. A knowledge-base which contained some basic information about blade-design was also constructed. Designs were

specified as feature elements --- for example, air-foil, shank, root, etc. in the case of the blade example. Each feature contains sub-features, thus complex products can be specified using this scheme. The user can select portions of a design (specific features) that is to be analyzed with the aid of this system. Such a facility enables partial (incomplete) designs to be analyzed thereby aiding concurrency.

An X-windows user interface for this program was built with the aid of the TAE+ workbench. This enables the user to select features of a product design and then subject it to a manufacturability test. The interface has provisions for data browsing, editing design parameters, report generation and explanation of its conclusions. The explanation process enables the engineer to determine the reasons why and how a particular design decision affected the manufacturability of the product. The system was implemented using LASER and TAE+ in the C language and was delivered for use on a Sun4 computer running a Unix operating system.

The Producibility Simulation aspect of this task was incorporated by developing knowledge-based simulation models of the casting and forging processes. The LSER/SIM discrete-event, knowledge-based simulation module was employed and elements of the casting process were identified and encoded. With this model in place, performance statistics can be computed which can be utilized in determining the manufacturability of the product. At the time, the only user interface that was released for Laser/Sim was based on SunView. This system was demonstrated during the December 15, 1989 and February 7-9, 1990 demonstrations held at CERC.

Conclusions:

The primary objectives of the Soft Prototyping Facility project were achieved by providing means for evaluating product designs for their manufacturability. By employing a feature-based design specification mechanism, certain short-cuts were utilized to circumvent the problem of identifying and process shape geometries. This approach enabled the primary thrust of the task to focus on design verification rather than to be bogged down in the morass of pattern-recognition. This is not to say that the feature identification process is unimportant, but it was considered more relevant by the researchers to focus

their efforts in the manufacturability issue and develop (or interface with) the feature-identification module at a later date.

A generic constraint-evaluation system was built that was fast and operated upon product design hierarchies. The system employed design data specified as Laser objects and which was also in the format of the simulation model. The ability to select portions of a design for evaluation purposes was found to be useful since it enabled incompletely specified (partial) designs to be processed --- encouraging concurrency of operations.

While the two modules --- the manufacturability evaluation and the producibility simulation -- were both implemented in C and used Laser objects for data representation, there was one issue that prevented integration --- the user interface. At the end of Phase 2, the Laser user interface used in the simulation models was based on SunView (since an X-based version of the Laser interface had not yet been released). However, the manufacturability evaluation module was implemented using TAE+ in compliance with the then-proposed suggestion that all user interfaces be built using TAE+. The resulting conflict in user interfaces prevented integration of the two modules.

Recommendations:

Follow-up work in this area should focus on the following areas:

- The manufacturability issue needs to be examined from the perspective of the entire design (composed of all its features and sub-features) rather than from its individual elements. Recommendations of manufacturability will then be valid for the overall product.

- The simulation interface should be upgraded to X-windows and integrated with the rest of the system.

- The system should be interfaced to operate upon data from CAD systems such as I-DEAS to facilitate use by engineers and

- There are numerous manufacturability issues for which constraint information or simulation models are not available. However, there are process planners in various domains. A truly comprehensible product

feasibility evaluation facility can be constructed by interfacing the SPF to such systems. By invoking the appropriate module, be it algorithmic, rule-based or simulation-model, an engineer needs to check out the complete viability of his design.

Publications:

The results of the research and development performed during this project have not yet been published. However, an abstract of a paper on this work is being reviewed by the DICE Program administrators prior to submission to a technical conference.

Hardware:

No computer hardware was purchased or developed in this task
No computer software was purchased in this task.
Two separate modules of software were developed during this task:

- a design evaluation tool that examined a design from the point of view of manufacturability of its features was constructed and

- simulation models were constructed for the casting and forging processes.

METHODS, TOOLS, AND ADVISORS FOR CE

DICE Program

Task 4.4.1.1 **Process Planning Advisor** **West Virginia University**

The process planning advisor task was organized into the following subtasks:

- Task 4.4.1.1.1 Volumetric Features(V_FACE)
- Task 4.4.1.1.2 Machining Process Analysis
- Task 4.4.1.1.3 Tool Path Optimization
- Task 4.4.1.1.4 Forging Process Simulation

Subtask 4.4.1.1.1 Volumetric Features (V_FACE)

Objectives:

The objective of this task was to implement V_FACE. V_FACE is a knowledge and geometry based advisory system for product design and process planning in concurrent engineering (CE) environments. It provides information such as product design features, process knowledge, tool path and optimal process sequence. It assists product design activity by providing production time and cost as well as general process knowledge. In addition, the user of V_FACE is able to interface with a variety of CAD and knowledge based systems.

Approach:

V_FACE was composed of 3 modules. The first module (V_FACE I) generated the volumetric features and surface features from geometry and topology information of product design. The manufacturing process knowledge for the features of the product such as machines, tools, parameters, production time and cost was generated by the second module (V_FACE II). This module also generated a tool path code and an improved geometry for each feature. The overall evaluation of the product was performed by the third module (V_FACE III) which generates optimal process sequence, process design knowledge and

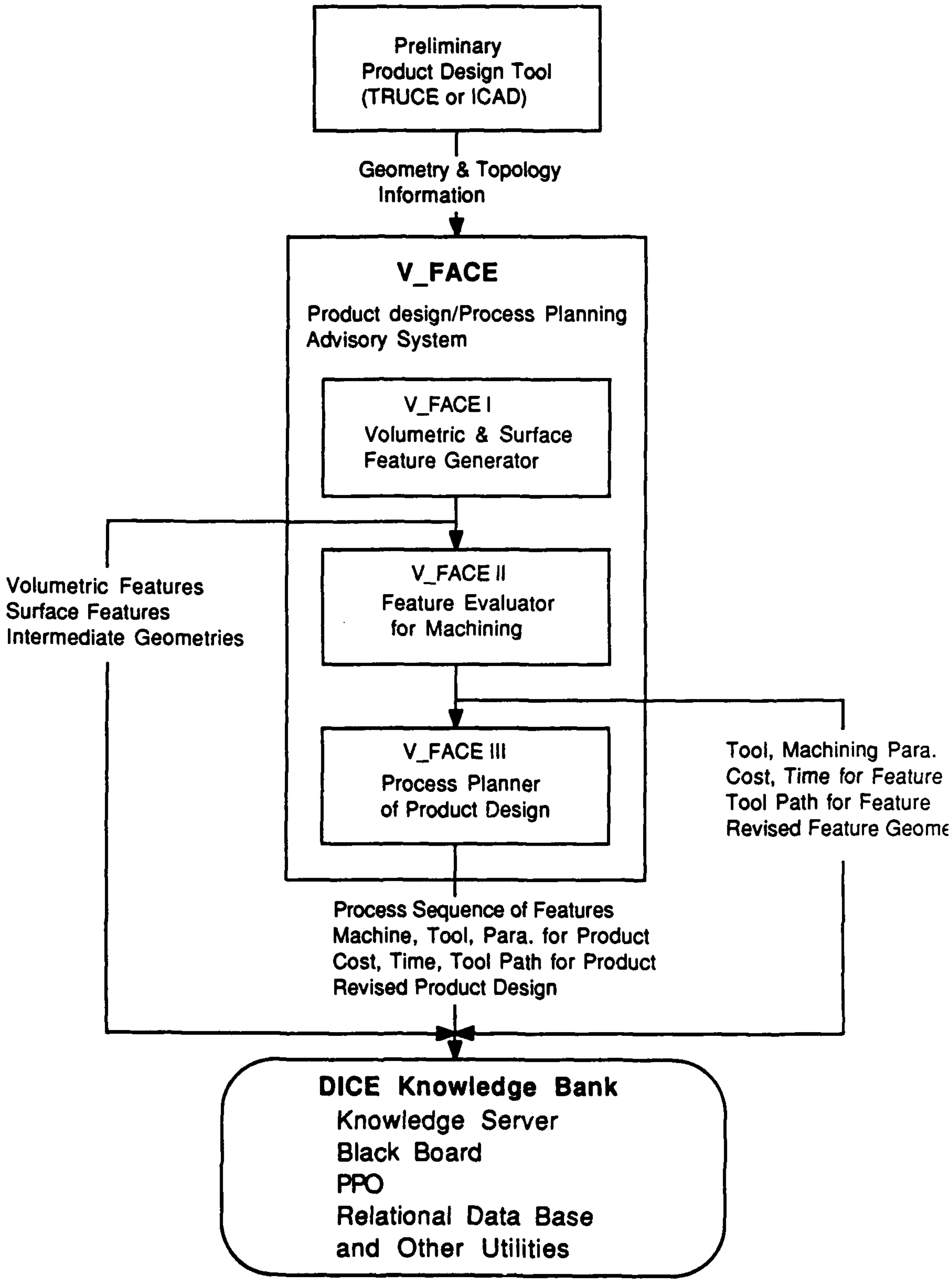
the final information for the CE service users. The data flow diagram for the overall system structure is shown on the next page.

Technical Results:

Previous work on feature generation was reviewed and summarized. The main research on the features had two basic approaches: one was to extract the features from the final model geometry and topology information and the other was to define the feature and design the model with those features. The feature extraction approach had some limitations with respect to a concurrent engineering environment. It does not provide enough information. In addition, the feature extraction process was inefficient. In this task, the feature information was redefined and efficient algorithms for feature processing were developed.
The basic algorithm for generation of volumetric features (V_FACE I) consisted of two parts -- low-level feature generation and high-level feature generation. The low-level feature generation was carried out by modifying the solid modeling procedure. The modeling procedure was represented by CSG tree and boundary representation.
As the application of the features (V_FACE II), the basic algorithm of feature based machining process evaluation was developed for the CE environments. The evaluation procedure involved operation selection, machine selection, tool selection, operation parameter selection and cost and time evaluation.

Conclusions:

A part design, like a turbine blade or a shaft can be created and displayed by a CAD system such as TRUCE. The V_FACE I module generates the necessary features of the part for the manufacturing process. These features or user specified features are evaluated for machining by the V_FACE II module. As a result of evaluation, machine, tool, machining parameters, time and cost are reported to the designer and process engineer. The V_FACE III module conducts trade offs of several features.


Preliminary
Product Design Tool
(TRUCE or ICAD)
Geometry & Topology
Information
V_FACE
Product design/Process Planning
Advisory System
V_FACE I
Volumetric & Surface
Feature Generator
Volumetric Features
Surface Features
Intermediate Geometries
V_FACE II
Feature Evaluator
for Machining
V_FACE III
Process Planner
of Product Design
Tool, Machining Para.
Cost, Time for Feature
Tool Path for Feature
Revised Feature Geome
Process Sequence of Features
Machine, Tool, Para. for Product
Cost, Time, Tool Path for Product
Revised Product Design
DICE Knowledge Bank
Knowledge Server
Black Board
PPO
Relational Data Base
and Other Utilities

Recommendations:

The current scope of V_FACE is to provide capabilities to evaluate the features for the machining process by the V_FACE II and V_FACE III modules. Future research will concentrate on such additional manufacturing processes as casting, forging and powder metallurgy. "System-X" will be developed to the point where customers can perform analysis on product design using System-X's syntax which is used for analytical and topological operation on geometric entities of a product as well as for inferencing with non-geometric entities.

Publications:

None.

Hardware:

The research team developed the following software modules:

1. V_FACE I (completed)
2. V_FACE II (partially)

Subtask: 4.4.1.1.2 Machining Process Analysis

Objectives:

The objectives of this subtask were to demonstrate the effectiveness of design for machinability, machining process selection, machine parameter determination and cost estimation for process planning in the concurrent engineering environment. Task members aimed to demonstrate the iteratively evolving design process at the advice of the above-mentioned modules, integrated and named the "Machining Advisor", pertaining to manufacturability. The nature of products considered included those with complex features as well as those with regular features. In order to produce designs which were easily machinable, it was necessary to develop a feature based design environment and to examine its potential. Since machining process selection was dependent on subjective variables and parameters, task members decided to develop expert systems to accomplish this function. Machine parameter estimation, as it is much dependent on the metal removal rate, was attempted with the use of machining data and statistical methods. The cost models were created using empirical formulae sensitive to the flow of information in the concurrent engineering environment.

Approach:

The general methods used to meet the above-mentioned objectives were unique to each module and were comprised of expert systems, special algorithms, statistical techniques costing methods etc.

- Feature Based Design Module

  Feature based design of a class of mating parts was accomplished using special algorithms which convert geometrical features into appropriate solid modeling command structures. FORTRAN was used as the programming language and TRUCE was used as the solid modeling environment.

- Machining Process Selection

  The selection of the milling process(es) for producing the design was performed by expert systems which acquired the geometric data on the part as well as data relating to the material, tool material and their properties.

- Machining Parameter Selection

  The machine parameters, including the feed, speed and depth of cut, were expressed as the metal removal rate and statistical methods were used to determine the relationships between work material hardness and the metal removal rate. FORTRAN and SAS were the principal tools used in the process.

- Machining Cost Estimation

  Empirical cost estimation models were developed and consisted of expressions to determine machining costs, tool costs, and set-up cost, . The estimated machine parameters were inputs to this module. FORTRAN was the primary tool used.

Technical Results:

The development of the above-mentioned modules gave rise to high-quality, publishable technical results. The development of the feature based module demonstrated the techniques that can be used to build large-scale, complex feature based design systems. Moreover, it provided a user friendly environment to use generally complex solid modeling packages. The feature based design tool can be used by designers to create products using manufacturing type geometric features and by manufacturing engineers to present modified designs to designers for consideration, using the same language of communication.

The expert systems developed for milling process selection provided substantial insights into methods for efficient inference, knowledge based design methods, knowledge acquisition for manufacturing and methods for object oriented knowledge representation. Machining parameter estimation

using statistical techniques, resulted in advanced methods for converting large amounts of experimental data into easily usable regression equations and showed the relationships between metal removal rate and the material hardness. The estimated metal removal rate was then used to determine the machine parameters.
The cost estimation models showed that tool costs and set up costs are relatively large compared to the machining costs and work material, tool material and their properties and geometry affect the total costs considerably. Information flow modeling within the system for concurrency also proved to be very effective in terms of use by the design stages.
The use of the overall system for the design and manufacturing of the class of parts considered showed that considerable reductions in product development times can be achieved by the interface between Machining Advisors and product design stages in the concurrent engineering environment.

Conclusions:

Research on the development of machining advisors of the nature mentioned above illustrated the need to examine their roles at various stages of design; namely, the preliminary and detail design. The nature of information, as it becomes available during the iterative process, ranges from scarce to complete. The Machining Advisors should be able to offer good advice at any level of design and with any amount of information. This prompted a study of the preliminary design of complex as well as regularly featured parts with respect to manufacturability. The intense study of the TRUCE solid modeling environment benefited us considerably and demonstrated the necessity to develop interfacial feature based design models.
Machine parameter estimation using statistical techniques proved that even in cases where the correlation between variables is poor, it is possible to integrate the use of algorithms and expert systems to alleviate the difficulty. In the overall sense, this activity showed the need for fundamental research to optimize the acquisition of data in an effective manner.
The cost estimation models proved useful in the evaluation of designs and showed insights into the development of cost optimization models which are necessary for the manufacturing engineer dealing with completed designs. The

results from this research also paved the way for analyzing machining cost optimization models for unisectional sculptured surface profile parts.

Recommendations:

The above-mentioned research established the effectiveness of machining advisors in the concurrent engineering domain. Future research needs to address the enlargement of the scope and complexity of the models described here. The models must also be sensitive to varying amounts of information as is the case during the iteratively evolving design process.

Machining Advisors need to be developed to address the evaluation of preliminary designs and to detail designs separately as each one is developed. The role of manufacturing will have to change from being passive to active. This means that modified designs need to be presented to the designer rather than merely being evaluated for manufacturability based on textual advice.

Publications

1. Gopalakrishnan, B. "Machining Advisor in Concurrent Engineering," Proceedings of the IIE Integrated Systems Conference, Atlanta, 1989.

Hardware:

The following software was developed:

1. Feature Based Design Software;
2. Machine Parameter Estimation Software;
3. Machining Process Selection Software and
4. Machining Cost Estimation Software.

Subtask 4.4.1.1.3 Tool Path Optimization

Objectives:

The overall objective of this task was to generate process plan from engineering design. In order to accomplish this objective, the process planning system needed manufacturing information. Some of the process planning information was extracted directly from CAD data. The focus of this subtask was to extract manufacturing features from CAD data and to develop an optimal tool path for machining application.

Approach:

I-DEAS, a CAD package, was utilized to generate the solid model. This solid model generates a universal file which has geometry and topology information of objects. The universal file also has the history of boolean operations. Since each data set of objects is blocked by -1 in the universal file, it was easy to follow the history of the object. Data of each object is composed of points, surfaces, nodes, leaf, etc. The data set of nodes gives information of each child generated and data set of leaves gives detail information and transformation matrix of each leaf. A Fortran program was developed to handle these data sets with dimensions of primitives such as cylinders, blocks, and spheres. The program identifies boolean operation and each child object. If an operation stands for boolean difference of a cylinder from a block on proper position, the feature generated by this operation is a hole. The boolean operation of differencing a block from another block generates a cavity such as a pocket, a slot or a step. This program followed major operation with primitive type and generated a list of manufacturing features. The manufacturing feature information was then utilized to generate an optimal tool path.

The optimal tool path was generated by minimizing travel distance of tools for producing manufacturing features. The standard tool path for each primitive was defined based on minimal travel distance. For a part which was composed of several primitives, a tool was temporarily selected for each primitive and a sequence of tool changes was generated based on the minimizing number of changes. The tentative tool motion was reviewed to globally minimize total

travel distance. An optimum number of tool changes were generated for a small set of tools. With tool change information, a sequence of machining operation was generated. This subtask generated a sequence of machining operation, tool change and tool positions. The input for this module is tool data and manufacturing features. The generated tool path was globally optimum for a given part. When the identified features were not of a standard shape, then design change was suggested. A data flow diagram for this task is shown on the next page.

Technical Results:

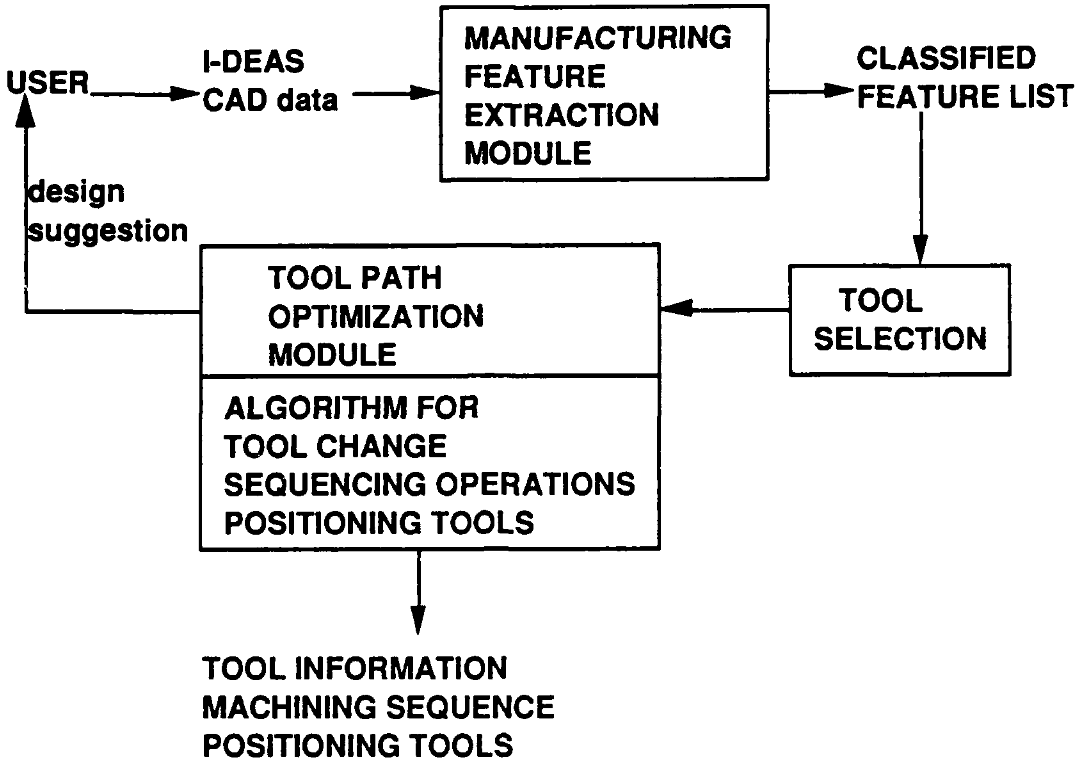
Since manufacturing requires knowledge of various processes, it is difficult to integrate CAD and CAM. This task generated a list of manufacturing features which contains feature type, dimension and position. The extracted features , an output of this task, included a hole, a pocket, a slot and a step. Since the generated feature did not consider a manufacturing process and did not have the knowledge required for processing, this information was not enough to support full integration of CAD and CAM. However, it can be used as part of CAD/CAM integration. The generated tool path was minimal for given basic primitives.

Conclusions:

This task generated features which are relevant for manufacturing. A high proportion of mechanical parts are composed of basic primitives such as pockets and holes. Complex features can be generated by operating an primitive feature and optimal tool path can be generated for the manufacturing features extracted from the CAD model.

Recommendations:

It is recommended that this task extract complex features from I-DEAS. The combinations of basic primitives can result in manufacturing features. The next task will use the combinations of primitive features to identify more complex



DATA FLOW DIAGRAM

parts. Based on complex features, minimal tool path generation will be expanded.

Publications:

None.

Hardware Developed:

The following software was developed:

1. Manufacturing feature extraction module and
2. Tool path optimization module (partially).

Task 4.4.1.1.4 Forging Process Simulation

Objectives:

The objective of this work was to develop and demonstrate an integrated methodology to design the forging process to produce turbine blades in a defect-free fashion. The process was analytically modeled using software tools based on finite element methods. Such analytical modeling is useful in identifying and optimizing the salient manufacturing parameters so that they can be controlled thus ensuring a defect-free product.

Approach:

The technical approach during this phase of the project was centered on using software tools based on rigorous numerical methods. Two nonlinear packages, NIKE2D and ADDS-Forming, were evaluated for their capabilities in simulating metal forming processes. Different process parameters considered in the analysis included: temperature, friction at the die and workpiece (billet) interface and material properties. The simulations aided in better understanding the influence of these parameters; this was useful for an improved production of forged, net shaped parts. With the help of these simulations, it was possible to accurately design the forging process for a particular product and to generate design rules with respect to forging of net shaped parts. This knowledge was incorporated into a knowledge base to provide an expert system. The finite element code was integrated with an expert system to provide a fully integrated package to evaluate the forging process for blade manufacturing.

Technical Results:

The isothermal forging process of shapes such as L-sections and H-sections was successfully simulated using the two finite element packages. In addition, the superplastic forming of thin sheets was conducted using these packages. NIKE2D is based on an elastic-plastic type of formulation to characterize the material behavior whereas ADDS-Forming is based on a rigid-viscoplastic type

of formulation. The material used in all the cases was Ti6Al4V and accurate data on the behavior of this material under deformation was provided.
It was found that ADDS-Forming is more suitable to conduct simulations of hot forgings whereas NIKE2D is more suitable for cold forming applications. This is due to the type of formulations employed and the convergence characteristics of the algorithms used to solve the highly nonlinear problems in these packages.
Since blade forging is conducted in the hot range, the ADDS-Forming package was used to simulate the blade forging process. In order to demonstrate the utility of such simulations in considering various design alternatives, different choices of preforms were used. Complete die filling of the metal was obtained during the simulations. The variation of the field variables across the workpiece cross section, was plotted and the load stroke curves for all the cases were generated.

Conclusions:

The use of software tools based on rigorous numerical methods is very useful for manufacturing process design. This is due to the accuracy of the results obtained which help the manufacturing engineer in designing the process efficiently. Analytical modeling of forging processes is a very viable alternative to the traditional build and test methods used in the industry to design processes. It should result in significant savings of material, time and manpower. Hence, further effort in this area is necessary so these technologies can become an important part of the concurrent engineering effort. However, the use of these packages is cumbersome and time-consuming because of the problems associated with the data input and the amount of core memory requirements needed for conducting the simulations.

Recommendations:

Currently, the capabilities of ADDS-Forming are limited to two-dimensional simulations. Hence, this package can be used for forging process simulations with plane strain and axisymmetric assumptions. However, such assumptions can lead to errors and do not truly reflect the complexity of the problem in the case of highly complicated shapes such as turbine blades. Therefore, it is

necessary to develop a software tool which can simulate the process in three dimensions. In addition, the process design activity conducted under this effort should be integrated with the product design effort conducted by other groups in the DICE program. This can be achieved by developing necessary software tools which can generate the die shape from the product geometry. Further, a robust preprocessor which can cater to the data handling requirements associated with the above-mentioned software tools should be developed.

Publications:

1. Dwivedi, S.N. " Computer Aided Simulation of H Section Forging Using ALPID", Submitted to ASME Journal of Engineering for Industry.

2. Dwivedi, S.N." Development of Constitutive Equations and Processing Maps of Inconel 718", Submitted in ASM Journal for material and Processing.

3. Dwivedi, S.N. and Shankar,R., " Computer Aided Simulation of H Section Forging using ALPID", ASME Conference on Intelligent Processing of Materials, 1990.

4. Dwivedi, S.N. and Shankar, R." Enhanced Forging Through Computer Simulation", Journal of Advanced Materials and Processes, pp 23-31, Feb. 1990.

5. Shankar, R. and Dwivedi, S.N. " Computer Aided Design for Manufacture of Rolled Rings," Accepted for publication in International Journal of Computer Integrated Manufacturing.

Hardware:

None.

DICE PROGRAM

Task 4.4.1.2 Geometric Modeling **West Virginia University**

Objectives:

(The objectives of this task are grouped into three numbered categories. The numbers are used throughout the report to refer to work connected with those categories.)

The three objectives included:

1. The development and testing of the product representation standards IGES and PDES; participation in the work of national organizations formed to pursue these ends. The development of software tools for interfacing mainstay industrial CAD packages, including TRUCE, IDEAS, PRO-ENGINEER and ICAD. This work addresses the technical problems posed to concurrent engineering efforts by the proliferation of differing geometric and product representation schemes as reflected in these and other CAD packages;

2. The development of improved curve and surface representations for geometric design and their associated algorithms and software. This work addresses several te' ·ical problems in geometric design: the need for rapid, interactive design of free-form curves and surfaces and geometric problems associated with these entities such as intersection calculation and offset generation for NC machining; and

3. New systems for handling complex geometric shapes based on the NOODLES system for describing and manipulating arbitrary non-manifold topologies; the development of graph-theoretic methods for feature recognition in such topologies. This work aims toward general, comprehensive approaches to geometric design by addressing problems inherent in current CAD systems.

Approach:

1. Representative task members became involved in the IGES/PDES community by attending meetings and serving on several national committees devoted to aspects of IGES/PDES development. This will allow CERC to become a leading player in ongoing and future standardization efforts. The objective of interfacing industry-standard CAD systems will be accomplished at CERC using the intermediary of translation to and from IGES files. Translation into the neutral data format will also be used. Effecting this translation will require detailed knowledge of these CAD systems' file formats; in some cases (e.g. TRUCE) some limited IGES translation software exists and will be utilized.

2. The work in curve and surface representation will employ two general approaches developed at CERC. The first approach, for box spline surfaces, is a method which allows for interactive surface design and local modification by manipulation of a grid of points in three-dimensions through which the surface is constrained to pass (referred to as box spline local interpolation). The second approach is that of polar spline curves in two and three dimensions,which have desirable properties of non-oscillation and for which offsets can be easily and reliably constructed.

3. The first step to be accomplished in using NOODLES-based topology in a CAD system is the incorporation of B-spline geometry into NOODLES. The approach to graph-theoretic methods for feature recognition will use surface-normal information, along with the topology of the graph to characterize and match surface features.

Technical Results:

1. GE software packages (TRUCE, POUT, RUFF, NC-VERIFY), as well as the ICAD system, were installed. IGES translators associated with these systems were tested using the IGES/PDES national organization's test case suite. In the December 15 Demonstration, intricate geometric data was successfully transferred from IDEAS to TRUCE through the

intermediary of an initial translation to an IGES file and then translation to a neutral database format.

2. Software development implementing the new box spline and polar spline curve and surface representation methods continued. For the box spline local interpolation method, software allowing for full three-dimensional manipulation of the surface was added onto the package completed in Phase I. New technical results on interpolation of tangent plane and curvature data were obtained and will appear shortly as a CERC technical paper. Polar spline software, including software for offset generation, is nearing completion. Some new results on offsets for, and approximation by, polar splines were obtained and were published in CERC technical report service.

3. NOODLES was installed on the Silicon Graphics Personal IRIS work station. Work linking B-spline geometry and NOODLES topology commenced. A new class of methods for feature recognition, combining information on surface normals with the graph-theoretic topological representation, was developed and a CERC technical report on these results is in preparation.

Conclusions:

1. Efforts undertaken by this task are leading to an active role by CERC in product representation standards development and indicate that CERC can, and should, become a prominent player in this area. These standards will play a key role in concurrent engineering. As shown in the December 15, 1989 Demonstration, the transfer of product information between CAD systems via IGES files is a viable component of this task and an important part of CERC's activities and the DICE vision.

2. The box spline local interpolation method is a useful tool for free-form surface development, representing a facility currently unavailable in CAD packages. Software development implementing this method has shown the feasibility of interactive free-form surface design based on local interpolation. The theoretical and software developments necessary to

make polar splines a useful entity for curve design are advancing; the facility for interactive curve design and automatic offsetting with polar splines will soon be available.

3. NOODLES is a powerful system for processing arbitrary non-manifold topologies. When combined with B-spline geometry and the new feature recognition algorithm for graphs of arbitrary topology developed under this task, the potential exists for CAD systems capable of performing required geometric calculations for very complex shapes.

Recommendations:

1. The ability of CAD systems to communicate with each other using a common language of product design is a key component in future efforts in concurrent engineering. As such, the emerging role of CERC in ongoing national efforts in product standardization should be supported and enhanced. Likewise, direct CERC efforts in designing interfaces between CAD systems as well as other research into IGES file generation and transfer should continue to be developed.

2. Implementation of the new representations and methods for curve and surface design developed under this task should continue. While these hopefully will ultimately find their way into future CAD systems, effort in the near term should proceed toward the development of software enhancements to current CAD systems. CERC expertise in IGES file generation and transfer obtained under this task can be effectively utilized in developing the required interfaces for these enhancements; this will further demonstrate the value of CERC's efforts in product representation standards.

3. Work aimed toward the development of CAD systems utilizing the powerful topology-processing capabilities of NOODLES should continue. Two components of this work should be incorporating B-spline geometry into NOODLES to allow for non-planar surfaces and the development of graph-theoretic algorithms for feature recognition which can be applied to the NOODLES topology representation. Incorporating B-splines will require research in curve and surface intersection as this is a major

component of CAD systems, and the complex topologies handled by NOODLES necessitate extremely reliable and efficient intersection algorithms.

Publications:

Papers

1. C.K. Chui and H. Diamond. "A general framework for local interpolation." TAMU CAT Tech. Report, June, 1989 (submitted for publication December, 1989)

2. H. Diamond. "Fundamental splines from spline spaces." CERC Tech. Report. , March, 1990 (to appear)

3. R. Lawson and C. Q. Zhang. "Alternative conditions for polar spline approximation." (in preparation)

4. W. W. Lai. "Polar-spline curve approximation algorithm." M.Sc. Dissertation, Department of Statistics and Computer Science, West Virginia University.

5. J. Faulkner and J. Miller. "B-rep Topology - an IGES entity specification." proposal.

6. G. Trapp. "IGES Software Data Capture Form." draft form.

7. C. Q. Zhang. "Polar spline approximation." Technical report TR88-10, Department of Statistics and Computer Science, West Virginia University; ARCHIVE-0023-89, Concurrent Engineering Research Center.

8. C. Q. Zhang. "Error analysis of polar spline approximation." Technical report TR89-2, Department of Statistics and Computer Science, West Virginia University.

9. C. Q. Zhang. "Polar-splines and offsets." Technical report TR89-3, Department of Statistics and Computer Science, West Virginia University; ARCHIVE-0022-89, Concurrent Engineering Research Center.

10. C. Q. Zhang. "Existence of offset." Technical Report ARCHIVE-0019-89, Concurrent Engineering Research Center.

11. C. Q. Zhang. "Polar-spline approximation." Technical Report ARCHIVE-0020-89, Concurrent Engineering Research Center.

12. C. Q. Zhang. "Proper offsets of closed convex curves." Technical Report ARCHIVE-0024-89, Concurrent Engineering Research Center.

13. C. Q. Zhang. "Intersections of curves and existences of proper offsets." Technical Report ARCHIVE-0025-89, Concurrent Engineering Research Center.

14. C. Q. Zhang. "Offsets of curves." Technical Report ARCHIVE-0026-89, Concurrent Engineering Research Center.

15. C. Q. Zhang. "Polar Spline in 3-Dimensional Spaces." (in preparation)

16. C. Q. Zhang. "Adjustable polar spline." (in preparation)

17. C. Q. Zhang. "Cylindrical spline approximation." (in preparation)

18. C. Q. Zhang. "Shape feature recognitions by face-relation-Based total graphs." Part One, "Representation and Recognition procedure." (in preparation)

Conference presentations

1. H. Diamond. "A General Framework for Local Interpolation and its Application." SIAM Conference, Tempe AZ, Nov. 1989

2. C.Q. Zhang. "Proper offsets of closed convex curves." Second National Symposium of Concurrent Engineering, Morgantown, Feb. 1990

Hardware:

Two preliminary versions of software modules which will run on a Silicon Graphics Personal Iris Workstation are nearly completed:

1. Box spline local interpolation module and
2. Polar spline curve design and offset module.

DICE PROGRAM

Task 4.4.2.4 Physical Models GEAE Lynn

Objectives:

The long term objective (i.e. Phase III) is to construct and demonstrate an integrated material behavior simulator which links process history, composite mechanics and life analysis. It will be compatible with the overall DICE system architecture and will be linked to the other major elements of the overall system. It will be accessible to materials developers and design engineers and will integrate the most current predictive models and material properties.

The specific objective (i.e. Phase II) was to demonstrate the feasibility of the simulator concept by designing a prototype simulator applicable for metal matrix composites (MMC's).

Approach:

The software and peripheral tools of the prototype simulator were developed under contract by Theta Systems, located in Woburn, MA. It uses a series of C-language programs interfacing with Microsoft's spreadsheet, EXCEL on a Macintosh computer. The user interface and control program is EXCEL.

Technical Results:

Based on the aforementioned approach preliminary software tools were developed and a user manual documenting the source code and operations was completed. The present capability of the simulator is limited to integrating two predictive behavior models, one of which depends on the output of the other for input. However, it demonstrates the functionality of the concept.

The software tools of the prototype simulator consist of the following:

a) a material properties database.
b) a library of models.
c) hierarchial rules that determine the proper sequence of model execution based on user input.
d) rules that map attributes and their models.
e) input and output rules for storage and retrieval of data.
f) a custom menu which enables input and output variables selection by user.
g) limited graphics capability to display output on a X-Y plot.
h) limited error checking capability on prescribed input variables.
i) ability to convert units from SI to English and vice-versa.

All of the above tools are preliminary and will require substantial modification/development in order to meet the long-term objective (ie. the "final" simulator).

Conclusions:

A full scale material behavior simulator could be developed using the experience and knowledge base of the prototype design. It would incorporate process history models, composite mechanics models and life prediction models, and thereby predict the full spectrum of MMC material behavior.

Recommendations:

Develop a 'final' full scale material behavior simulator in Phase III.

Publications:

None.

Hardware:

None purchased against DICE funding.

DICE Program

Task 4.4.4.2 Engineering Models (Part A) West Virginia University

Objectives:

The objective of this task was to analyze the influence on natural frequencies and mode shapes as changes to blade geometric and stiffness characteristics for any given profile of turbine blades. Coriolis acceleration, centrifugal forces, warping effects and rotary inertia were considered.

Approach:

A large-deflection, small-strain Euler-Bernoulli beam formulation was used for the finite element analysis. The spatial discretization of the beam was carried out by employing a special beam element with fifteen degrees of freedom. A cubic spline interpolation method was developed to prescribe the profile of airfoils. Related section properties were calculated by means of numerical integrations. The dynamic analysis was performed in dimensionless quantities such that the deflections and natural frequencies were normalized by the length of the blade and the rotational speed of the turbine, respectively.

Technical Results:

Three sets of results were generated. The baseline turbine blades considered had a length of 30", chord of 3", cross section of NACA 0009 and three typically unsymmetric airfoils which were made of aluminum and rotating at 10,000 rpm. The first set of results investigates the effect of changing the thickness to chord ratio on natural frequencies and mode shapes. The second set shows how natural frequencies vary with taper ratio while keeping the area of the blade constant. The final set of results indicates the trend of the natural frequencies with the increase in twist rate.

Conclusions:

The results show that the change of thickness had very little effect on the rotating natural frequencies and modal shapes of the blade. However, the natural frequencies decreased with an increase in the taper ratio. The variation of the tip twist angle and first torsion mode shows that the effect of higher twist was a reduction of the natural frequencies.

Recommendations:

Further investigations considering the effects of shear deformations of turbine blades should be carried out since, in most cases, the shear deformation can not be neglected when blades with small aspect ratios are studied.

Publications:

N.T. Sivaneri and Y.P. Xie. "Automatic generation of free-vibration characteristics of pretwisted turbine blades with given profile." 1990 (in preparation).

Hardware:

A Fortran computer program was developed by the authors for utilization of the dynamic analysis.

DICE Program

Task 4.4.4.2 Engineering Models (Part B) **West Virginia University**

Objectives:

Objectives of the task were:

1. To synthesize the modeling strategy to produce finite element models of complex mechanical parts based on the engineering parametric definition of the geometric features of the part;

2. To produce a finite element model for analysis purposes which reflects the type of analysis required and the presence of loads and boundary conditions and also the degree of refinement needed for the particular application at hand;

3. To assess the parametric sensitivity of the stress function of complex parts as well as the frequency response and aerodynamic performance with respect to design parameter variations;

4. To characterize the influence that changes of basic blade profile configurations may have on the natural frequencies and mode shapes of the airfoil section of a blade;

5. To develop optimum design rules for composite ducts and cylinders;

6. To develop generic models for the analysis of composites laminates with interlaminar defects;

7. To develop an integrated material simulator for optimum parametric design of stiffened ducts and shafts and flanges made of laminated composites; and

8. To advise the DICE user of the necessary structural modifications required to achieve a desired modal behavior.

Approach:

1. The approach for Point 1 above was the use of solid constructive geometry to generate a solid model of the part using the parameters which define the geometry of the part and then to apply algorithms for finite element mesh generation through logical procedures that incorporate modeling rules to render an efficient finite element model.

2. The approach used for Point 2 above was structured into a software module called KATY-PARFEM, which stands for Parametric Finite Element Modeling. This module uses an approach based on FEM meshing algorithms and generic rules for stress oriented or dynamic oriented modeling practices.

3. For Point 3 above, the least squares curve-fitting was used to develop relationships representing four objective functions: maximum root stress, natural frequency, material cost and aerodynamic performance. Numeric differentiation was performed with central differences, and Newton's method was used for one-dimensional searches, while the method of feasible directions was used for multiobjective optimization.

4. For Point 4 above, a large-deflection small-strain Bernoulli beam formulation was used. The spatial discretization was done with a 15 DOF beam element. Cubic splines were used to prescribe the airfoil profile and sections properties were obtained through numerical integrations. The analysis was carried out in normalized terms to produce non-dimensional relationships.

5. For Point 5 above, curves were developed that show the relationship between number of plies and fiber orientation angle for various loading conditions, including internal pressure, bending moment and transverse shear. The algorithm incorporated the following failure criteria: maximum stress, quadratic and compressive criterion, bending and torsional buckling criterion. A suitable factor of safety could be input by the user for specific applications.

6. For Point 6 above, the modified Donnell approach was used, with stresses obtained from classical engineering and correction factors that satisfied the equilibrium compatibility conditions for two dimensional elasticity. This approach

incorporated the transverse shear and normal strain, both of which are generally neglected in the classical theory.

7. For Point 7 above, the approach used consisted of two components -- one which provided an estimation of a composite flange performance and one which optimized its configuration in terms of the design parameters which defined the laminate lay-up patterns. Classical laminate field theory and failure criterion were used in these modules.

8. For Point 8 above, modeling was performed using the GEOMOD and SUPERTAB modules of I-DEAS, which used to be performed using substructuring methods in ANSYS, but was now performed using the SYSTAN module of I-DEAS. The accuracy of natural frequencies and the amount of CPU time were selected as performance metrics for the MODFEM module.

<u>**Technical Results:**</u>

1. For Point 1 above, a computer program was developed which takes the engineering parameters of a turbine blade as input and automatically produces a solid model of the turbine blade configuration with one command.

2. For Point 2 above, a computer program was developed which uses the solid model generated in the previous point and transforms it into a finite element model which reflects a degree of refinement that is appropriate for stress analysis and for dynamic analysis. This model was then made available for stress and dynamic performance evaluation.

3. For Point 3 above, the results obtained with finite element models were compared with beam and plate theories as variations on the parameters were taken. Good comparisons were obtained for thin chords and lift loading. Centrifugal loading produced stress concentrations at both the leading and trailing edges.

4. For Point 4 above, three sets of results were generated for turbine blades with a length of 30", chord of 3", cross section of NACA 0009 and three unsymmetrical airfoils which were made of aluminum and rotated at 10000 rpm. The first set of results investigates the effect of changing the thickness to chord ratio on natural frequencies and mode shapes. The second set shows how the natural

frequencies vary with taper ratio while keeping the area of the blade constant. The final set of results indicates the trend of the natural frequencies with the increase in twist rate.

5. For Point 5 above, the minimum wall thickness of a composite duct or pressure vessel which satisfies the failure criteria for a broad range of loading and geometric parameters was presented for three different types of lay-ups of unidirectional plies: angle ply (0/90) and (0/90 & +45/-45). The optimum fiber orientations, lay-up patterns and the corresponding minimum wall thickness were studied with respect to variations in the individual load components, geometric parameters and material properties.

6. For Point 6 above, predictions from the model were compared with those of a FEA solution. All the cases, for which comparable results were available, exhibited agreement. The model was capable of predicting the stresses and displacements much more accurately, especially in the regions of high stress concentration.

7. For Point 7 above, an on-line iterative composite flange design program was developed which allows the designer to select different types of composite materials, lay-up patterns, butt-joint effect, dimensions of the flange and loading conditions for optimum parametric design of laminated composite flange. For demonstration purposes, four lay-up patterns of 0/90, 30/-30, 45/-45 and 0/45/-45/0 were analyzed under identical loading conditions. It was found that lay-up 45/-45 with gore angle of 30 degrees is the optimum choice for making the laminated composite flange. To verify the accuracy of the numerical predictions, a series of coupon specimen tests was also conducted. It was found that the experimental results agree with the numerical predicted values.

8. For Point 8 above, the accuracy of the natural frequencies and the amount of the CPU times were most affected by the number and location of Master DOF and by the number of modes used in the structural analysis.

Conclusions:

1. For Point 1 above, the main accomplishment was to reduce the analyst modeling time by providing a tool that directed the solid modeler into creating a solid

model, with the features of the part being modeled using the engineering parameters which define its configuration.

2. For Point 2 above, the major contribution was the reduced effort required to generate a finite element model that takes into account the degree of refinement, the type of analysis and the type of loads required in the design assessment analysis.

3. For Point 3 above, the developed software will be useful in identifying critical design features and provides appropriate design rule guidance for the designer.

4. For Point 4 above, the results show that the change of thickness has very little effect on the rotating natural frequencies and modal shapes of the blade. However, the natural frequencies decrease with a increase in the taper ratio. The variation of the tip twist angle and first torsion mode shows that the effect of higher twist is to reduce the natural frequencies.

5. For Point 5 above, for angled ply lay-up, the optimum weight of the duct is obtained when the angle is kept between 45 and 55 degrees (approximate). Nearly the same optimum (minimum) duct thickness was obtained with all three lay-up patterns. Buckling was found to be the most important concern while designing ducts for the optimum weight.

6. For Point 6 above, the developed code is expected to be useful in the study of interlaminar defects in composites. The model is especially suited when a number of configurations are to be evaluated. Two separate computer codes were developed to analyze laminates with or without interface defects. The analytical models developed assume frictional effects to be present over areas of delamination where a frictional relationship exists between the normal and shear stress components at the interface.

7. For Point 7 above, an interactive laminated composite flange design computer program was developed. Using laminate plate theory and failure criterion, this program can be used for on-line optimum design of laminated composite flanges. Four design examples are presented to demonstrate the adaptation of the concurrent engineering approach for design cycle reduction. Furthermore,

coupon specimen tests were conducted to confirm the validity of the design results.

8. For Point 8 above, selection of the number and location of Master DOF needs considerable designer experience in order to effectively modify the modal characteristics of the model. Use of master DOFs that are optimally selected is the key to successful reduced-order analysis.

Recommendations:

In order to assess different design alternatives and the performance of various systems (mechanical or structural), the development of engineering models is essential. However, the quality of the results obtained in the simulation of the systems behavior depends heavily on the quality of the model from the analytical standpoint. Ultimately, the quality of the actual design will depend on how well the system was represented through meaningful engineering models.
Effort was aimed towards the development of efficient models which reflect not only the degree of refinement required by the particular application but also the environment for their development. This exercise resulted in:

1. Design rules applicable to specific cases from parametric studies including turbine blades, ducts, shafts, and flanges;

2. Software which is capable of advising the user in the creation of complex models with a minimum of effort, such as in the case of PARFEM and MODFEM; and

3. Methodologies which can be applied to generic cases to produce models, design rules and parametric studies. These methodologies in engineering modeling represent the core of what is being implemented into the DICE architecture as part of the engineering analysis workstation activities.

Finally, it is recommended that the programs and results thus far obtained be used to illustrate two fundamental aspects needed for DICE success -- namely the generic nature of the methodologies developed and the capability of providing specific cases for the testbed currently under development.

Publications:

- Kang, B. S.-J., J.Prucz, and F.Hsieh. "Parametric Design of Laminated Composite Flange." To be presented in Fifth International Conference on CAD/CAM, Robotics and Factories of the Future, Norfolk, VA, December 2-5, 1990.
- Mucino, V. H., J.E.Sneckenberger, J.C.Benner, and S.Chung. "Parametric Finite Element Modeling Techniques in Concurrent Engineering Environments." Proceedings of the Second National Symposium on Concurrent Engineering, Feb 7-9, 1990.
- Prucz, J., and M.Lambi. " A Practical Engineering Approach For Predicting Interlaminar Stresses in Composites." To be presented in Fifth International Conference on CAD/CAM, Robotics and Factories of the Future, Norfolk, VA, December 2-5, 1990.
- Ruth, M. K. "Parametric Modeling and Analysis of Turbine Blades for Concurrent Engineering Environments." Masters Thesis, Department of Mechanical and Aerospace Engineering, WVU, May 1989.
- Sivan, J., J.Prucz and P.C.Upadhyay. "Parametric Design of Composite Ducts and Pressure Vessels." To be presented at ASME Winter Annual Meeting, Dallas, TX, November. 25-30, 1990.
- Sivaneri, N. T., and Y.P.Xie. "Automatic generation of free-vibration characteristics of pretwisted turbine blades with given profile." 1990 (in preparation).

Hardware:

Hardware purchased includes:

- Silicon Graphics 4D Series Workstation;
- Sun-4 Workstation;
- DEC VaxStation3200;
- Macintosh II; and
- IBM PS/2.

DICE PROGRAM

Task 4.4.5.1 XD Material Characterization GEAE

Objective:
The objective of this task was to characterize selected material properties of XD titanium aluminide castings. Special emphasis was given to the mechanical properties of importance to turbine blade design. These included tensile, creep, low cycle fatigue, fracture toughness, and impact energy.

Approach:
As reported previously for Phase 1, the XD Ti-48Al-2V+7.5% TiB_2 cast bars used in this task were acquired from Howmet Corporation. The cast bars had the dimensions of approximately 14 x 0.7 x 0.9 inches, and represented three different casting conditions. They were 1) centrifugal casting with a 600F mold preheat temperature, 2) centrifugal casting with a 1200F mold preheat temperature, and 3) static casting with a 1200F mold preheat temperature. Throughout this report, these three casting conditions will be referred to as Casting Condition A, B, and C, respectively. The cast bars were HIP'ed by Howmet at 2300F/25 ksi for 4 hours to close casting porosity and heat treated at 1650F for 16 hours in vacuum. All test specimens were fabricated from the cast bars in longitudinal orientation.

As planned, testing was conducted for smooth and notched tensile, creep, low cycle fatigue(LCF), fracture toughness, Charpy impact, thermal expansion, dynamic modulus and cyclic oxidation. The smooth bar tensile tests were conducted at RT and 1600F with the tensile strains monitored by extensometry attached directly on the gage section of the test specimens. The notched tensile tests were conducted at RT, 1200F and 1600F on circumferentially notched bar specimens. The nominal stress concentration factor (Kt) of the notched specimens was 2. The creep tests were carried out at 1600F in laboratory air. The creep strains were measured by extensometry attached to the gage section of the test specimens. The LCF tests were performed at RT, 1200F and 1600F in a strain-control mode for the strain A-ratio (alt./mean) of infinity and the test frequency of 0.33 Hz. For fracture toughness testing, a chevron-notched short rod specimen geometry was used per ASTM E1304. The nominal dimensions of the short rod speciemns tested were 0.5 inches in diameter and 0.75 inches long. The short rod tests were conducted at RT and 1200F. For impact testing, instrumented Charpy v-notch tests were performed at RT, 1200F, and 1600F. The

dynamic modulus tests were carried out in the temperature range of RT-1500F, and the thermal expansion tests were conducted for 200F-1800F. Finally, cyclic oxidation tests were performed at 1600F and 1800F for 100 hours in laboratory air. The oxidation test cycle used consisted of a 15-minute heating, 1-hour hold at test temperature, followed by a 6-minute forced air cool to below 200F. A summary of the test conditions investigated in this task is shown in Table 1.

Metallurgical evaluations were performed on selected specimens to understand failure mechanisms and the effects of microstructure on mechanical properties. Optical and scanning electron microscopy was employed for these evaluations.

Technical Results and Discussion:
The tensile test results obtained from smooth bar specimens are listed in Table 2. The results for the three XD TiAl casting conditions were virtually the same. This is shown, for example, in Figures 1 and 2 for ultimate tensile strength (UTS) and plastic elongation, respectively, for room temperature. The average observed UTS and elongation for XD TiAl at RT were 88.6 ksi and 1.1 %, respectively. For comparison, the RT UTS for cast Rene'77 is 152.6 ksi and the tensile elongation for cast Rene'41 is 5.0 %. At 1600F, the average observed UTS and elongation for cast XD TiAl were 62.7 ksi and 19.6 %, respectivley. These can be compared with 95.0 ksi for UTS of cast Rene'77 and 16.0 % for elongation of cast Rene'41 at 1600F. It can be noted in Table 2 that the tensile elongation of the specimen 8-24 from Casting B is unusually higher (36.53%) than those of other specimens tested at the same temperature. However, the tensile reduction of area of this specimen was just about the average. A replicate test may be required to determine if this observation is truly representative for the casting.

The notched tensile test results are listed in Table 3 and plotted in Figure 3 as a function of temperature. It was observed that the notch tensile strengths (NTSs) of XD TiAl remained almost constant at approximately 91 ksi in the temperature range RT-1200F, and increased to about 98 ksi at 1600F. This was in contrast to the smooth bar tensile results which showed a reduction of UTS by 30 % with increasing temperature from RT to 1600F. As a result, the notch strength ratios, as determined by the ratio of NTS to UTS, for XD TiAl increased from a notch brittle/ductile borderline 1.0 at RT to a notch ductile 1.5 at 1600F.

TABLE 1
TESTS CONDUCTED FOR CAST XD TITANIUM ALUMINIDE

TEST TYPE	CASTING CONDITION	TEMPERATURE, F				NUMBER OF TESTS
		RT	1200	1600	1800	
Tensile (Smooth Bar)	A	2	-	2	-	4
	B	2	-	2	-	4
	C	2	-	2	-	4
Notched Tensile	B	2	2	2	-	6
Creep	A	-	-	3	-	3
	B	-	-	3	-	3
	C	-	-	2	-	2
Low Cycle Fatigue	A	1	2	1	-	4
	B	3	2	3	-	8
Fracture Toughness	A	1	1	-	-	2
	B	3	3	-	-	6
Charpy Impact	A	2	2	2		6
	B	1	1	1		3
Dynamic Modulus	A	1	1	-	-	1
Thermal Expansion	A	-	1	1	1	1
Cyclic Oxidation	A	-	-	1	1	2
	B	-	-	1	1	2

TABLE 2

TENSILE (SMOOTH BAR) RESULTS FOR CAST XD Ti-48Al-2V+7.5%TiB_2

TEMPERATURE (F)	CASTING CONDITION	SPECIMEN NO.	0.02% YS (KSI)	0.2% YS (KSI)	UTS (KSI)	PLASTIC ELONGATION[a] (%)	REDUCTION OF AREA (%)
RT	A	7-05	44.7	69.2	89.7	1.11	3.3
RT	A	7-12	47.5	70.8	88.0	1.00	3.4
RT	B	8-03	49.5	69.4	88.5	1.04	3.3
RT	B	8-04	49.3	71.3	90.4	1.07	3.0
RT	C	9-05	46.9	69.3	89.6	1.22	3.8
RT	C	9-07	48.0	68.1	85.1	0.95	3.2
1600	A	7-14	24.5	48.3	61.7	12.56	31.6
1600	A	7-15	30.2	47.8	61.7	16.04	32.6
1600	B	8-24	32.1	49.3	62.9	36.53	30.9
1600	B	8-25	32.1	50.4	64.0	(b)	23.0
1600	C	9-06	35.6	49.1	61.7	15.92	33.1
1600	C	9-08	34.1	49.5	64.2	16.82	32.0

(a) As determined from extensometer measurement or from strain rate subsequent to yield.
(b) Extensometer slipped; % E not available.

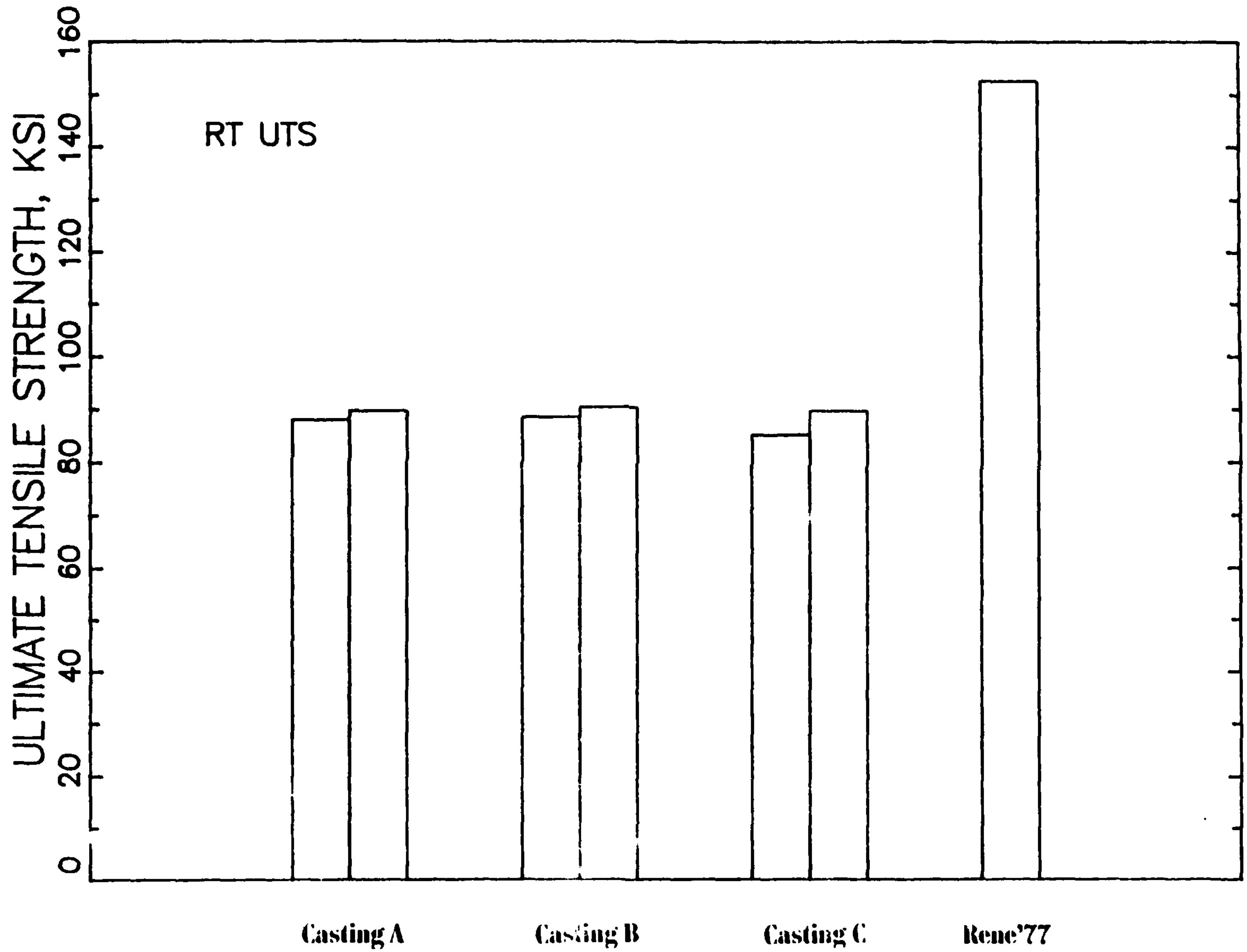


Figure 1 Room Temperature Tensile Strengths of Cast XD TiAl and Rene'77.

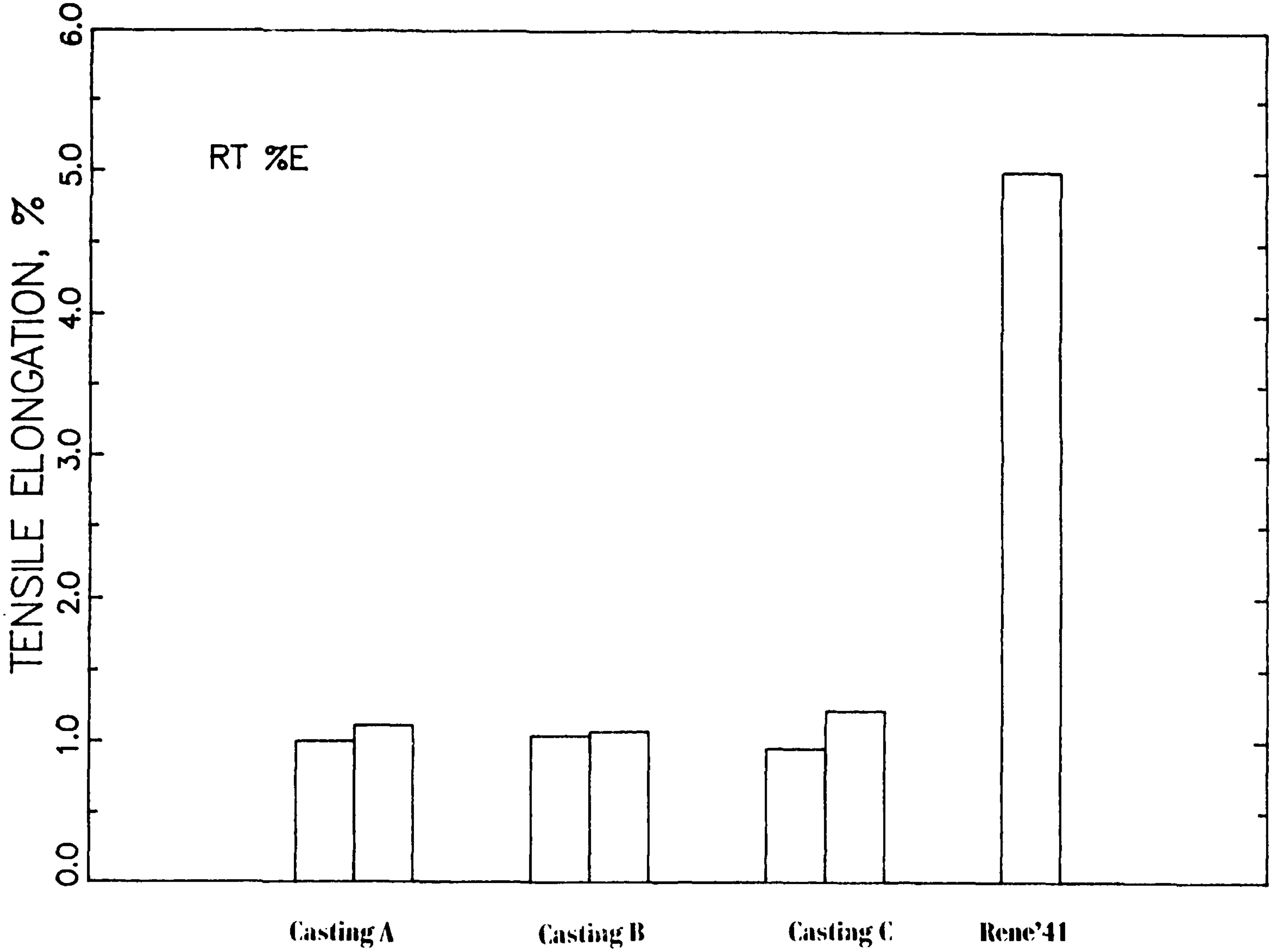


Figure 2 Room Temperature Tensile Elongation of Cast XD TiAl and Rene'41.

TABLE 3

NOTCHED TENSILE RESULTS FOR CAST XD Ti-48Al-2V+7.5%TiB_2

TEMPERATURE (F)	CASTING CONDITION	SPECIMEN NO.	NOTCH TENSILE STRENGTH, NTS (KSI)	RATIO NTS/UTS
75	B	8-15	90.2	1.01
75	B	8-18	94.4	1.06
1200	B	8-19	88.9	-
1200	B	8-16	91.9	-
1600	B	8-20	97.6	1.54
1600	B	8-17	98.6	1.55

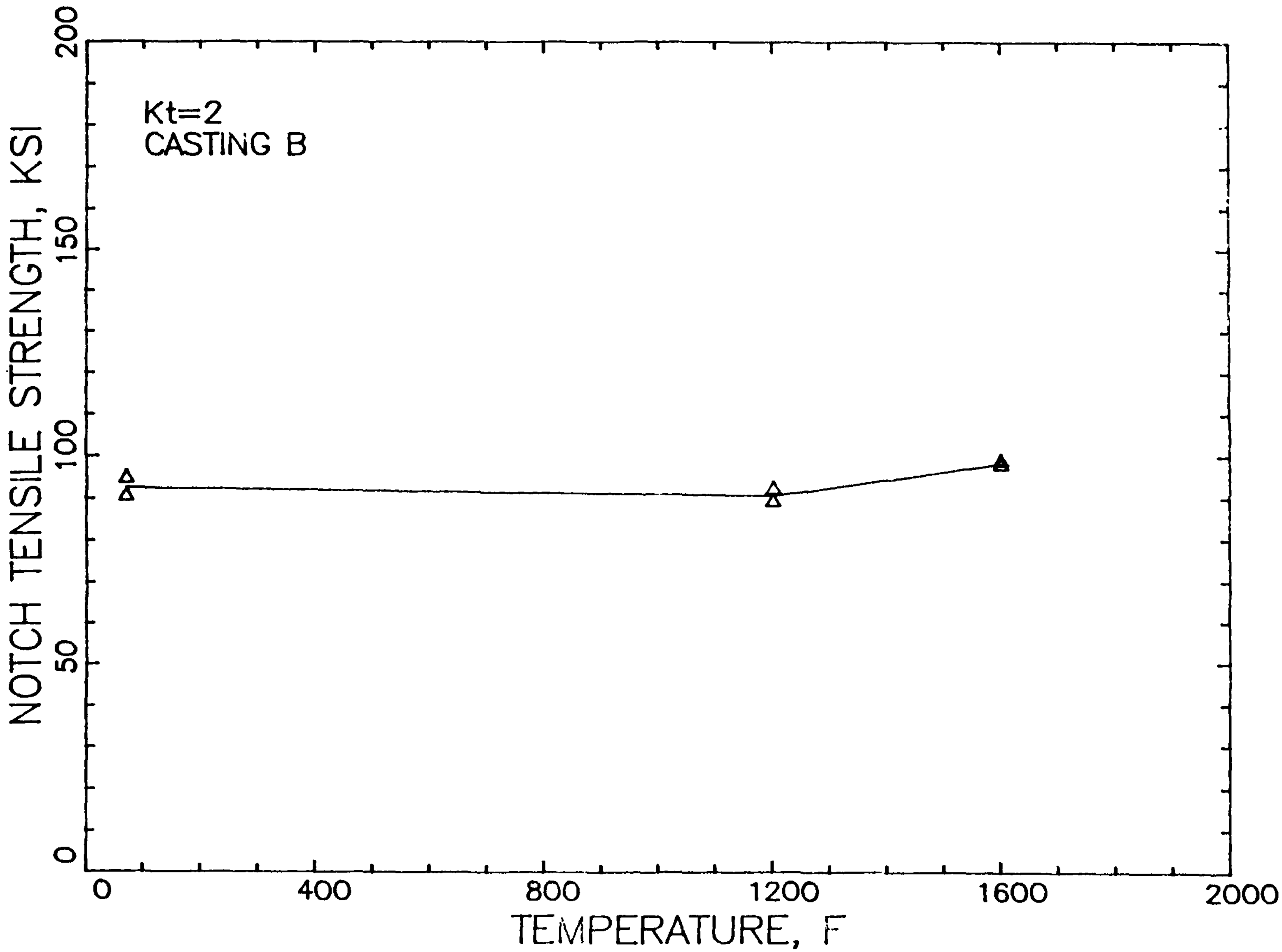


Figure 3 Notch (Kt = 2) Tensile Strengths of Cast XD TiAl for RT-1600F.

The creep test results are listed in Table 4 and the hours to reach 0.2 % creep strain at 1600F are plotted as a function of stress in Figure 4. No significant differences were observed in creep capability among the three casting conditions, Fig. 4. The general appearances of the strain-time creep curves obtained from the XD TiAl specimens were quite typical of many conventional alloys, exhibiting the usual primary and steady state creep regions. Under the testing conditions investigated, the primary creep strain averaged at about 0.15 %, and the steady state creep rates varied from approximately $1\text{x}10^{-4}$ to $1\text{x}10^{-5}$ per hour. As compared with cast Rene'77, the 0.2 % creep strengths of XD TiAl were significantly lower, Figure 5.

The LCF test results are listed in Table 5 and plotted in Figure 6 as a function of total strain range vs. cycles to falure for RT, 1200F and 1600F. Here again, no significant differences were observed in LCF life among the three casting conditions tested. The LCF lives were generally independent of temperature in a short life regime, about 1000 cycles or less. In an intermediate life regime, 1000-10000 cycles, the test results for 1200F were higher than for both 1600F and RT. The fatigue failure modes in the tested specimens were observed to be quite similar for all three temperatures. As shown in Figure 7, a single fatigue origin typically occurred on the specimen surface and propagated transgranularly along a crack plane normal to the loading axis. Note in Figure 7 that the fracture surface actually consists of lamella patterns, which was apparently caused by cracks poropagating along the interface of gamma and alpha-two phases in transformed lamella grains. Although some TiB_2 particles were occasionally observed in fatigue origin areas, the origins generally did not appear to be caused by any material defects or microstructural constituents. The observed LCF lives for XD TiAl were comparable to those for cast Rene'77 and Rene'41 at all three temperatures. Figure 8 shows one such comparison for the LCF life at 1200F.

Fracture toughness test results are listed in Table 6 and plotted in Figure 9 as a function of temperature for Casting Conditions A and B. Overall, the average fracture toughness values at RT and 1200F were 15.2 amd 20.6 $\text{ksi(in)}^{1/2}$, respectively. The test results from Casting A were approximately 10 % higher, on average, than those for Casting B. The fracture modes, however, did not appear to be different for these two castings. For both castings, fracture proceeded transgranularly in a quasi-cleavage mode at RT and along the interface of gamma and alpha-two lamella structures at 1200F. Typical fracture surfaces for RT and 1200F are shown in Figure 10.

The cyclic oxidation test results for XD TiAl showed that the final weight changes in four samples tested at 1600F and 1800F

TABLE 4

CREEP RESULTS FOR CAST XD Ti-48Al-2V+7.5%TiB_2

TEMP (F)	STRESS (KSI)	CAST COND	SPEC NO.	TIME 0.2% (hr)	TIME 2% (hr)	PRIMARY CREEP[(a)] (%)	STEADY STATE CREEP RATE (hr^{-1})	TEST DURATION (hr)	FINAL CREEP (%)
1600	10.0	A	7-04	6.9	150.0	0.15	1.3E-4	153.7	2.049
1600	10.0	A	7-11	5.9	157.0	0.20	1.1E-4	179.8	2.277
1600	10.0	B	8-05	9.0	201.0	0.16	9.2E-5	234.7	2.309
1600	10.0	C	9-03	5.1	147.0	0.25	1.2E-4	166.2	3.6
1600	5.0	A	7-10	49.0	-	0.13	1.6E-5	200.1	0.450
1600	5.0	B	8-26	59.0	-	0.13	1.5E-5	200.3	0.421
1600	5.0	B	8-27	47.0	-	0.14	1.5E-5	201.2	0.44
1600	5.0	C	9-04	61.0	-	0.08	2.3E-5	200.3	0.518

(a) As determined from the intercept of the tangent at steady state creep rate with the strain axis at zero time.

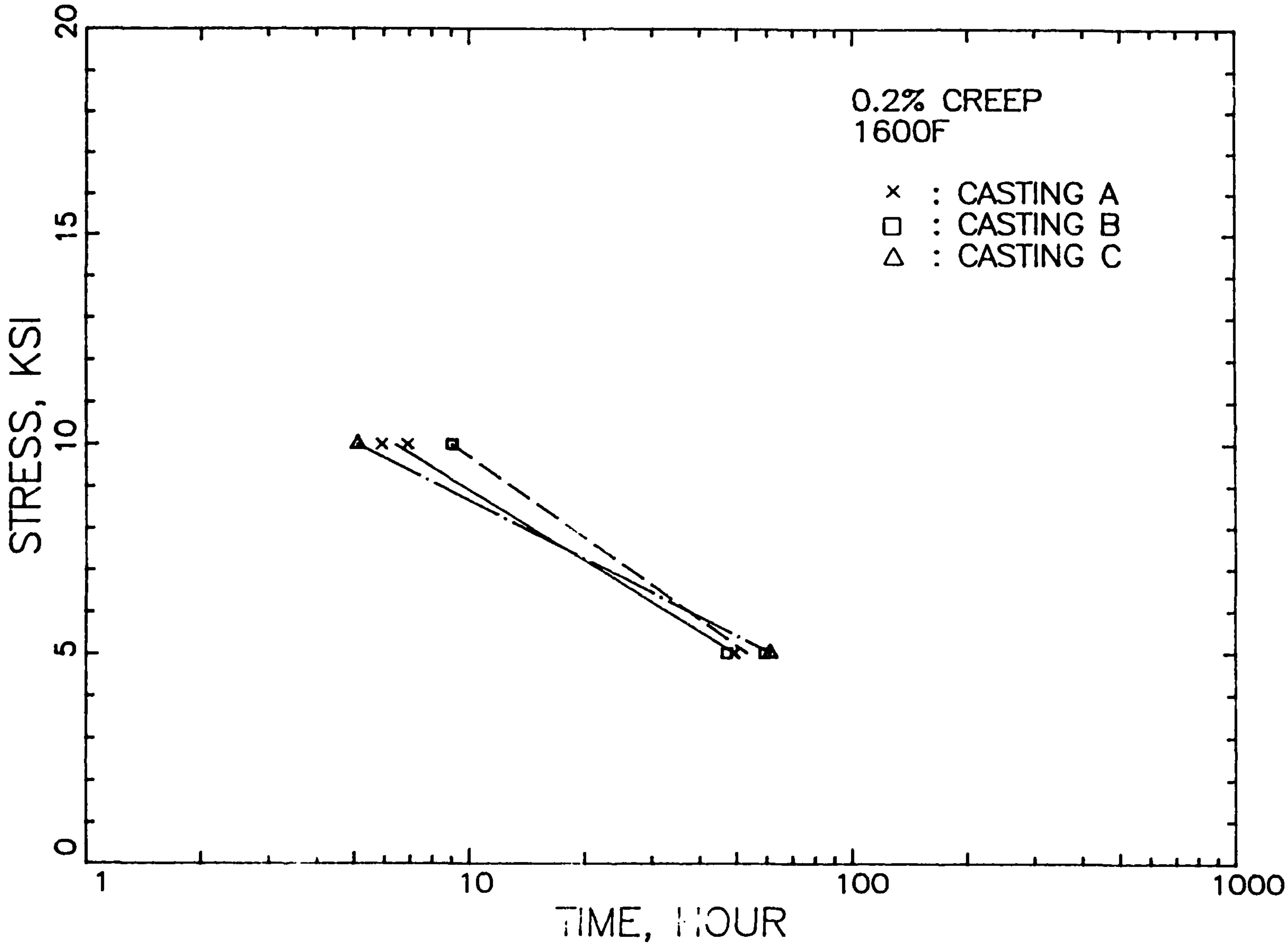


Figure 4 0.2% Creep Life of Cast XD TiAl at 1600F.

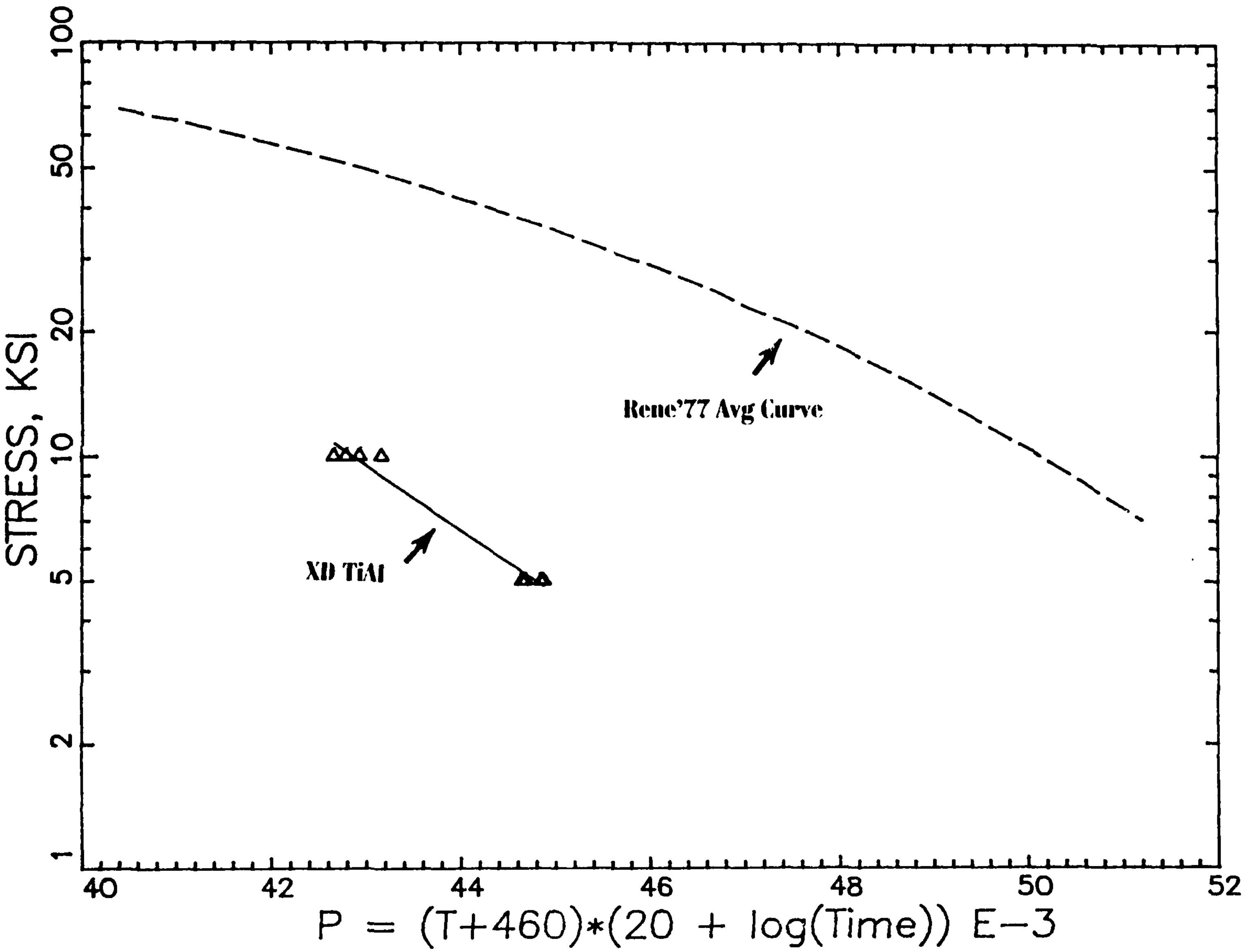


Figure 5 Comparison of 0.2% Creep Strengths Between Cast XD TiAl and Rene'77

TABLE 5
LCF RESULTS FOR CAST XD Ti-48Al-2V+7.5%TiB_2
Strain-controlled, 20 CPM, A-ratio=Infinity

TEMP (F)	TOTAL STRAIN RANGE (%)	CASTING CONDITION	SPECIMEN NO.	ELASTIC MODULUS (x1E6 PSI)	ALT. PSEUDO STRESS (KSI)	AT HALF LIFE MAX STRESS (KSI)	AT HALF LIFE MIN STRESS (KSI)	AT HALF LIFE PLASTIC STRAIN (%)	CYCLES TO FAILURE
78	1.0	B	8-01	26.8	134.0	92.0	-98.2	0.29	659
78	0.8	A	7-01	26.4	105.6	75.5	-76.9	0.22	281
78	0.6	B	8-13	26.3	78.9	69.1	-61.6	0.10	2,970
78	0.45	B	30-01	27.1	61.0	62.4	-53.9	0.02	104,573+
1200	1.5	B	8-14	23.7	177.8	79.6	-82.2	0.82	38
1200	1.0	A	7-03	22.1	110.5	69.0	-67.3	0.38	463
1200	0.8	B	8-02	23.9	95.6	61.2	-67.5	0.26	2,435
1200	0.6	A	7-13	18.5	55.5	53.8	-42.1	0.08	100,000+
1600	1.02	B	8-09	18.6	94.9	53.9	-54.2	0.44	321
1600	0.8	B	30-02	21.7	86.8	53.1	-56.0	0.30	674
1600	0.6	A	7-07	19.7	59.1	45.7	-43.2	0.15	4,239
1600	0.45	B	8-21	22.0	49.5	39.7	-38.4	0.10	10,949

Note: "+" after cycles to failure denotes runout.

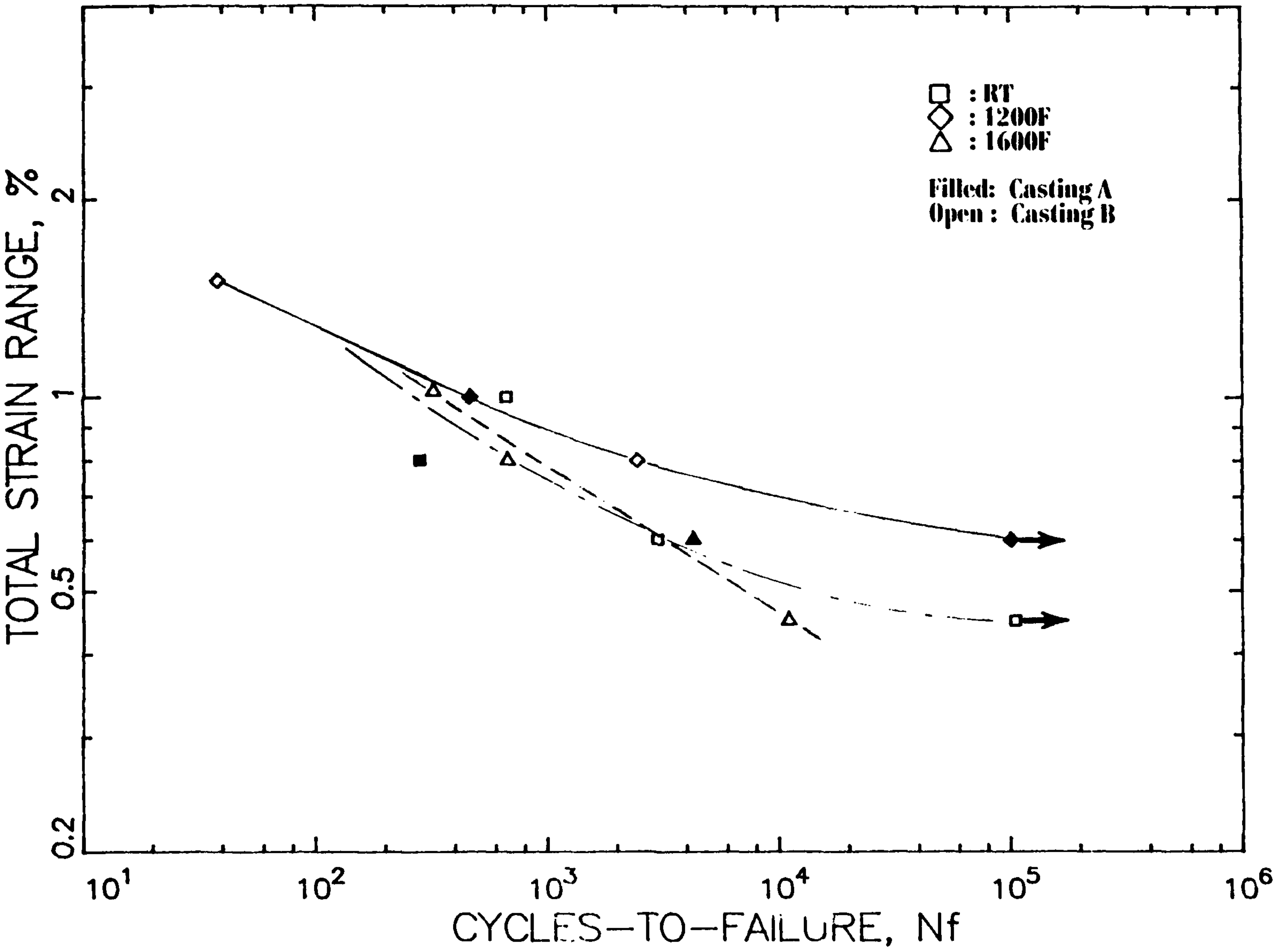
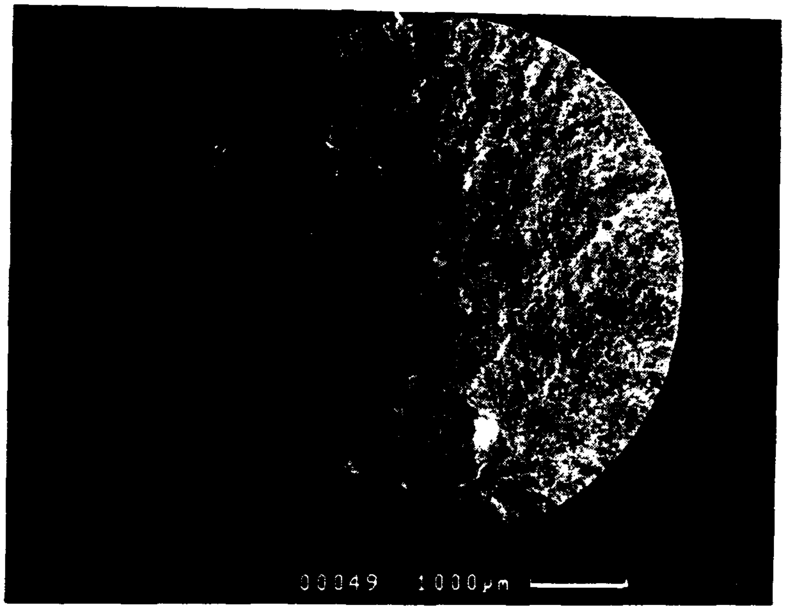
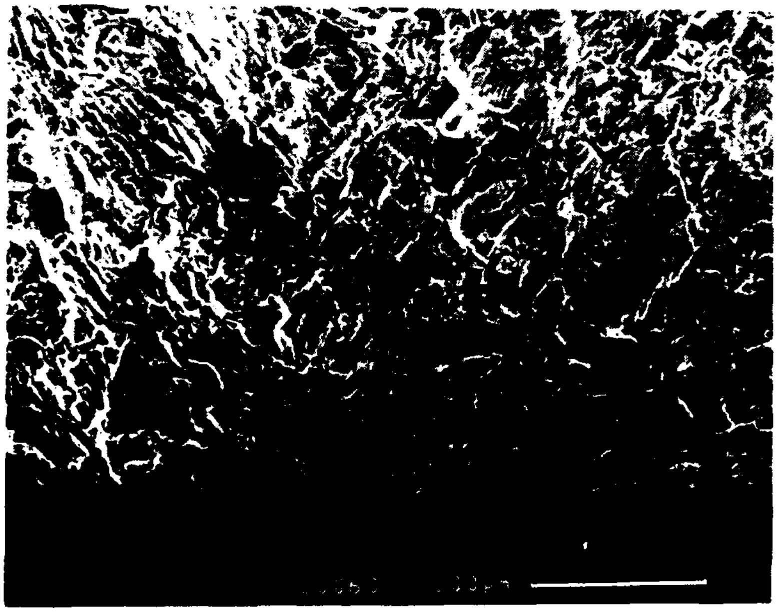


Figure 6 LCF Life of Cast XD TiAl for RT-1600F.



a)

Fatigue Origin



b)

Figure 7. SEM Fractographs of Cast XD TiAl LCF Specimen Tested at 1200°F, Showing a) Overall Fracture Appearance, and b) Fatigue Origin Area. (Specimen No. 8-02)

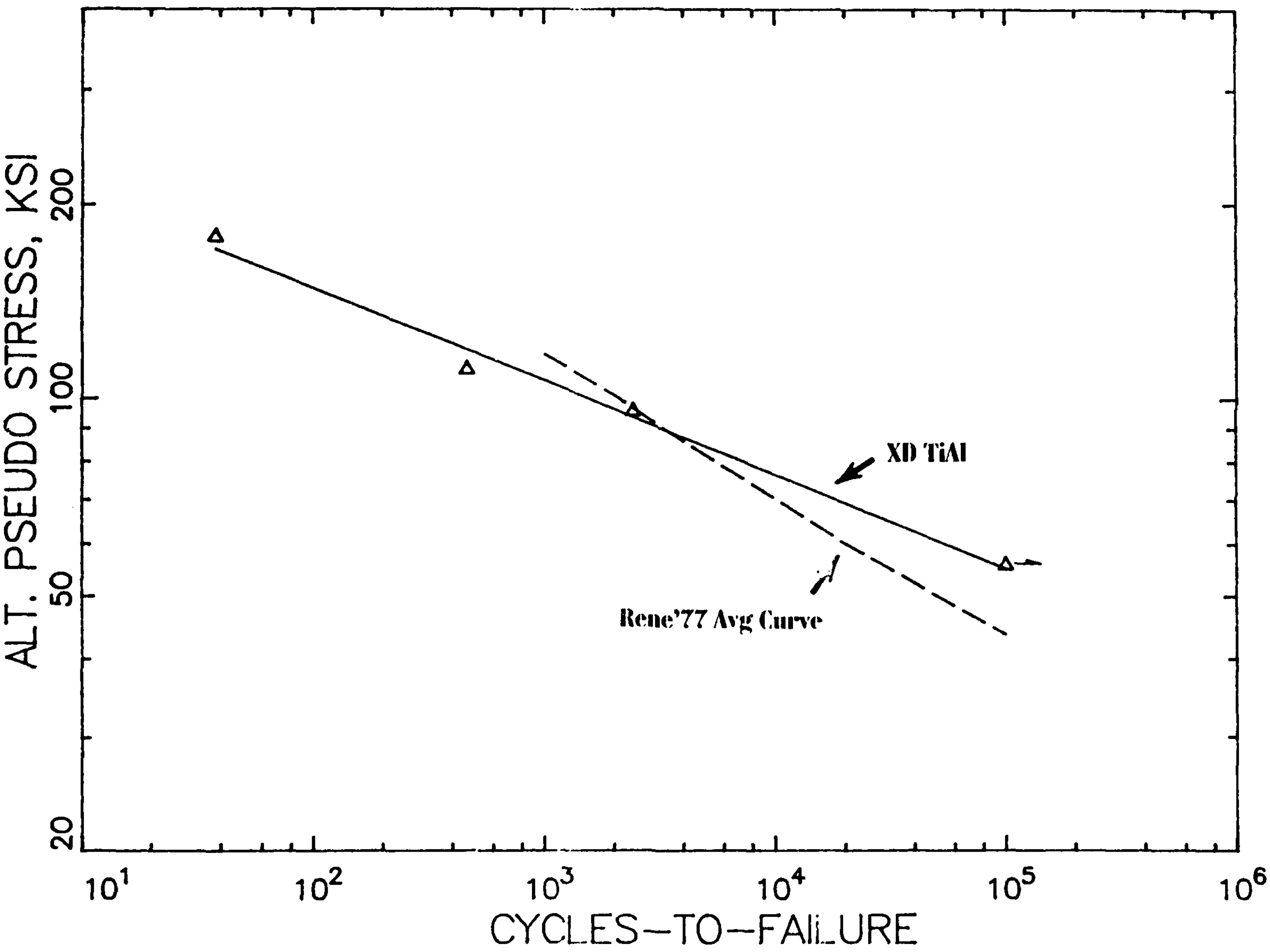


Figure 8 Comparison Between Cast XD TiAl and Rene'77 For LCF Life At 1200F.

TABLE 6
FRACTURE TOUGHNESS RESULTS FOR CAST XD Ti-48Al-2V+7.5%TiB_2

TEMP (F)	CASTING CONDITION	SPECIMEN NO.	FRACTURE TOUGHNESS K_{IV} (KSI*SQRT(IN))	FRACTURE TOUGHNESS K_Q (KSI*SQRT(IN))	VALID PER ASTM E1304?	REASON FOR INVALIDITY	"P" FACTOR
70	A	7-17	16.25	16.25	Yes	-	0.082
70	B	45-04	12.39	12.39	Yes	-	0.030
70	B	45-06	15.79	15.79	Yes	-	0.017
70	B	8-22	16.54	16.54	Yes	-	0.036
1200	A	7-18	-	22.27	No	High "p"	0.242
1200	B	8-23	-	18.82	No	High "p"	0.165
1200	B	45-07	20.19	20.19	Yes	-	0.083
1200	B	45-05	-	20.99	No	High "p"	0.121

Note: The maximum allowed "p" or Plasticity factor is 0.1 per ASTM E1304.

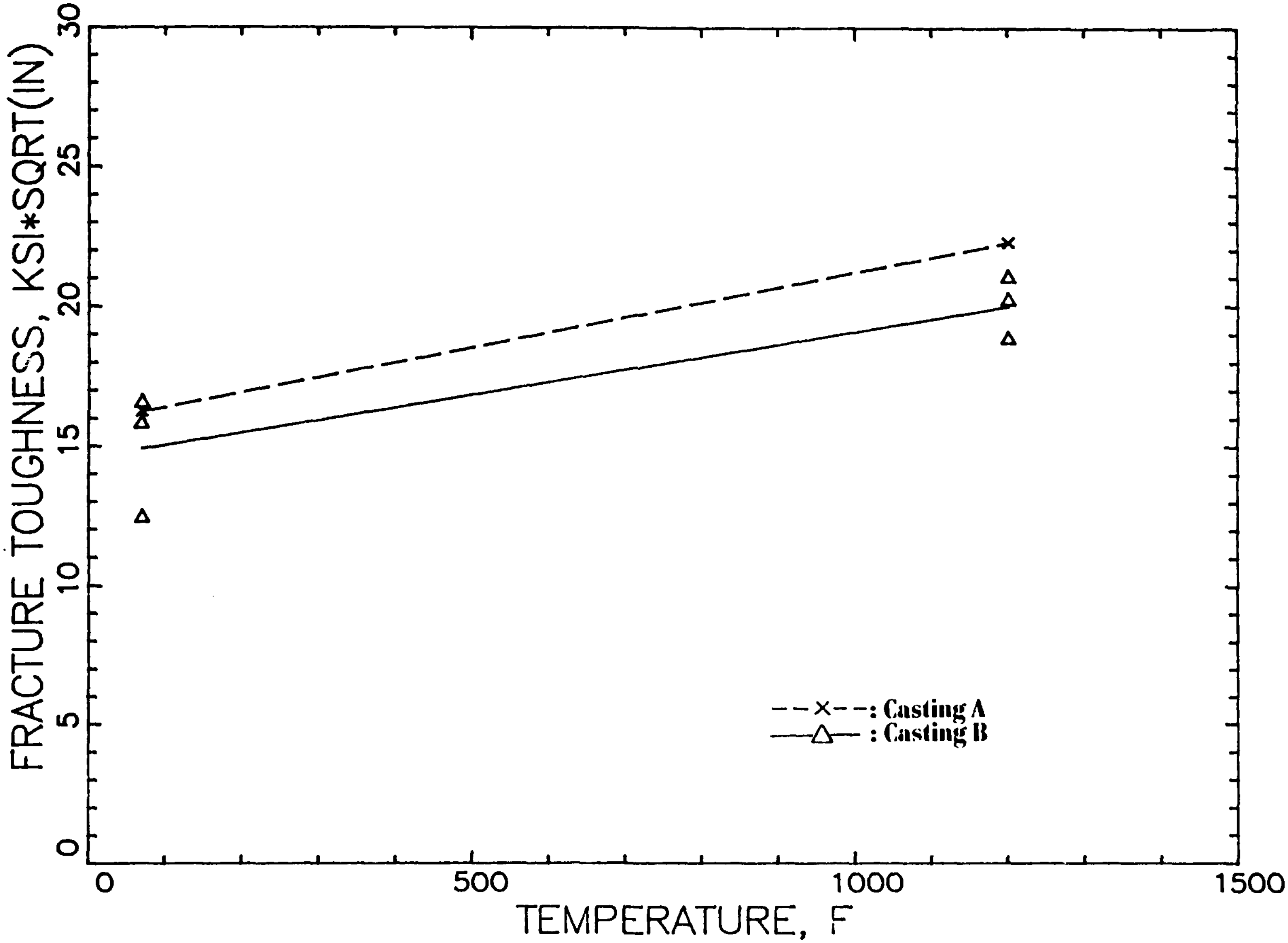
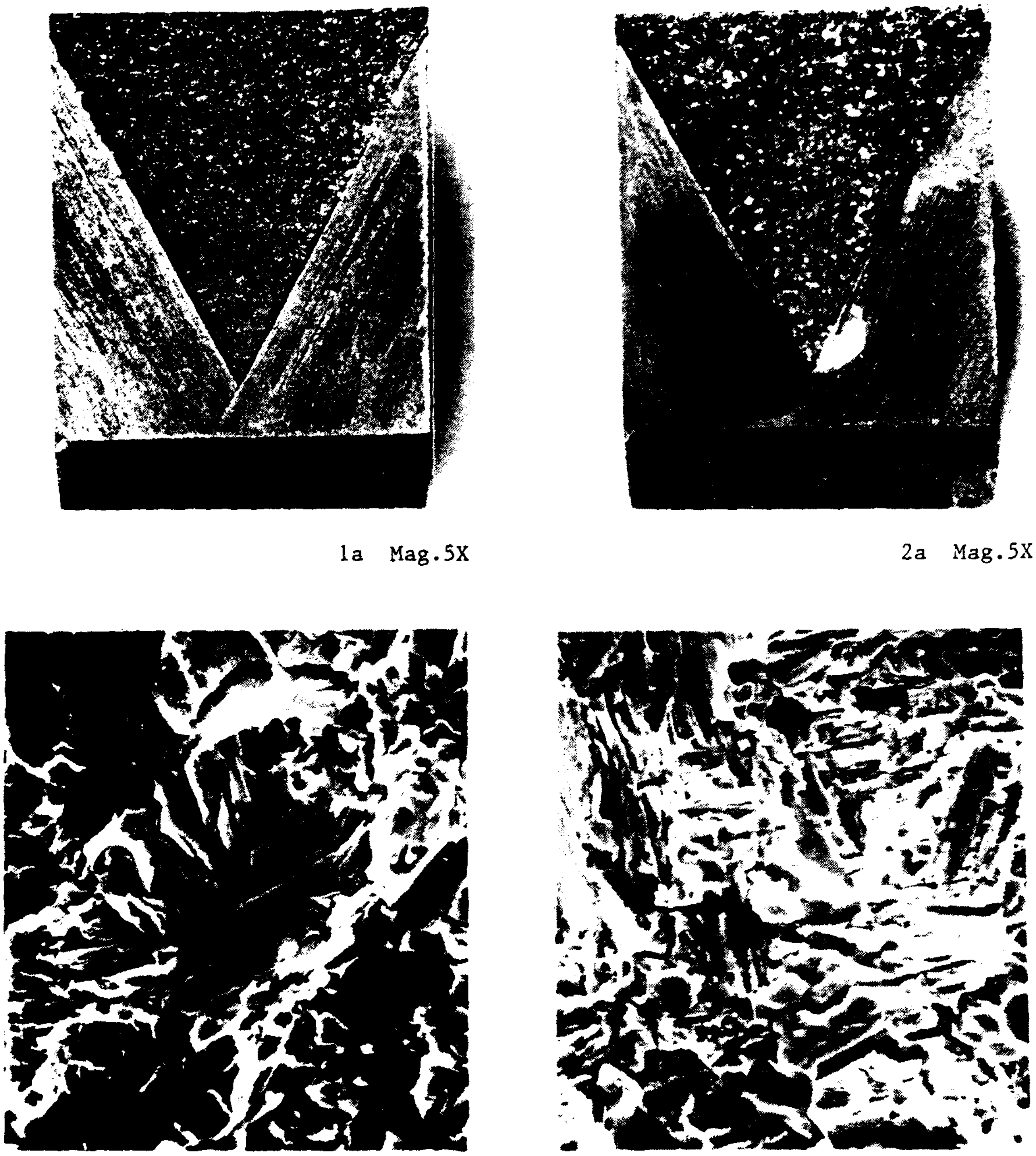


Figure 9 Fracture Toughness of Cast XD TiAl at RT and 1200F.



1a Mag.5X

2a Mag.5X

1b Mag.1000X

2b Mag.1000X

Figure 10. Fractographs of XD TiAl Chevron-Notched Short-Rod Fracture Toughness Specimens Tested at RT (1a and 1b) and 1200°F (2a and 2b), Showing Overall Fracture Appearances and Fracture Details at Chevon Center.

were -9.1, -14.07, -78.25, and -79.39 mg/cm^2. At each temperature, two samples each representing Casting Conditions A and B were tested for 100 hours. However, due to apparent mix-up of test samples, these test results could not be positively identified with individual samples. A replicate testing should be performed to determine the effects of casting condition and test temperature on cyclic oxidation resistance of XD TiAl.

The test results for Charpy impact energy, thermal expansion, and dynamic modulus are the same as previously reported for Phase 1. It was shown in Phase 1 Report that the Charpy impact energy for cast XD TiAl varied from approximately 0.2 ft-lbf at RT to 0.36 ft-lbf at 1200F, and that Casting A had slightly higher impact energy values than Casting B in the temperature range RT-1600F. It was also reported that the mean thermal expansion coefficients of XD TiAl were approximately 20 % lower than cast Rene'80, and that the dynamic moduli for XD TiaAl were lower than wrought Inconel 718 at temperatures below 1200F but higher at temperatures above 1200F.

Conclusions:

Key mechanical properties were obtained for XD TiAl castings to determine, in part, the effects of casting parameters on properties. The casting parameters investigated were a) centrifugal casting with a 600F mold preheat temperature, b)centrifugal casting with a 1200F mold preheat temperature, and c) static casting with a 1200F mold preheat temperature. These three casting conditions were observed to have only modest effects on fracture toughness and Charpy impact energy but no apparent effects on tensile, creep, and LCF. This suggests that some other paremeters such as chemical composition and microstructure may play a more important role for these properties.

As compared with nickel-base cast alloys such as Rene'77 and Remn'41, the mechanical properties for cast XD TiAl were significantly lower, especially the creep strength. One exception was LCF. The LCF capability of XD TiAl appeared to be comparable to that of Rene'77. In addition, the observed low thermal expansion coefficients and dynamic modulus can be advantageous to XD TiAl under thermal fatigue conditions, although this same low dynamic modulus can be disadvantageous in HCF.

Overall, much improvements in mechanical properties are desired for cast XD TiAl. These include, in particular, RT tensile ductility, creep, and fracture toughness. The XD TiAl castings,

however, were otherwise sound, free of porosity, and relatively uniform in microstructure.

Recommendations:

In light of the test results obtained for XD TiAl in Phase 2, the following additional efforts are recommended:

- Improve creep strengths of XD TiAl by chemistry/microstructure modifications.
- Further investigate the effects of casting conditions on fracture toughness and Charpy impact energy.
- Conduct additional cyclic oxidation testing to positively determine the oxidation resistance of XD TiAl castings.

Publications:

No paper was published during Phase 2, using DICE funding.

Hardware:

No hardware or software was purchased or developed under this task.

DICE PROGRAM

Task 4.4.5.2 ICPD Materials **GE-CRD**

OBJECTIVES

Because of their high fabrication temperatures and differences in coefficients of thermal expansion, components made of metal matrix composites develop significant residual stresses during the fabrication process. Consequently, the tasks of material property definition as well as mechanical design are closely related to the fabrication process. Effective application identification and component fabrication will require that issues associated with these three areas of materials, processing and design be treated concurrently.

One of the Phase 1 goals of this task focused upon the micromechanical issues associated with these materials. Models to predict thermally induced residual stresses which result from different coefficients of expansion for the fiber and the matrix were developed. These residual stresses also have an influence on the mechanical behavior of these composites and the models can address the responses to fundamental, composite thermomechanical loads.

However, in addition to the micromechanical level, there are a number of other residual stress related problems which had not yet been addressed. Specifically, if the composite component requires a cross-plied laminate, residual stresses will be induced because of differences in the coefficients of expansion in the two principal lamina directions. As a result, the laminating process and its related residual stresses will have an effect on component mechanical response. These effects must be understood if effective design methodologies including optimization techniques are to be developed and applied to this class of materials.

The specific objective of this task was to combine modeling routines developed under

Phase 1 DICE funding to allow residual stress and mechanical response predictions to be made for laminated metal matrix composites. This combined model, RESLAM, has been completed. Its capabilities and limitations are discussed in this final report. Examples of several results are also presented.

APPROACH

The general approach adopted here assumes a simple model incorporating the most significant physics of the problem will be most effective in supporting the parameter studies and trade-offs necessary during the early stages of the development of a new material/component system. Detailed modeling conducted on other programs is used for guidance in formulating the simple models developed here. These models are consistent with the requirements of optimization software which is being developed under another task of this program.

For the specific focus of Phase 2, the individual programs developed and written under Phase 1 have been combined in a fashion that allows constitutive material properties and laminating sequence to be simultaneously considered with regard to their effects on both residual stresses, mechanical loads and thermal loads. Hashin's bounding models (discussed in the Phase 1 final report) are used to predict the orthotropic stiffness properties of an individual lamina. Standard laminate analysis used to support optimization calculations, also reported in Phase 1, is used to define the stresses induced in the individual laminae due to thermomechanical loading. Finally, a simple micromechanical model accounting for weak interfacial normal strength is used to determine micromechanical residual stresses and failure criteria.

TECHNICAL ACCOMPLISHMENTS

The software developed here allows for a new type of parameter study to be carried out within the context of concurrent material and component development. The significant issues of residual stresses and weak interface effects which have been identified within the context of unidirectional metal matrix composites can now be explored in a generally laminated form also. The discussion which follows will highlight the assumptions, limitations and operation of this software and provide several examples of results.

Assumptions

Three models are combined to accomplish the goals outlined here. Hashin's equations are used to predict the orthotropic stiffness properties of a composite lamina as a function of the constituent material properties. These equations assume that the fiber and the matrix are transversely isotropic in nature. The approach utilized by Hashin results in upper and lower bounds on the properties due to the generally irregular and unknown geometric arrangement of the fibers in the composite. The relative proximity of the bounds will vary depending on the properties of the constituents making up the composite. The equations are elastic in nature.

Standard laminate analysis is used to calculate the stresses induced in the composite during cooling from process temperature as well as due to thermomechanical engineering loads. This approach is also elastic in nature and assumes plane sections remain plane during the deformation process. It is a plane stress solution and assumes that there are no stresses through the thickness of the composite. Within the context of the linked system of models used here, this model uses the lamina stiffness properties predicted by the Hashin equations

along with a description of the laminating sequence of the composite and the thermomechanical loads to calculate the stresses in each of the composite plies. Whenever there is a difference in the coefficients of expansion for the two principal directions of a composite lamina, residual stresses will be induced at the ply level as the the laminated composite is cooled. These stresses are calculated before any engineering loads are applied.

Finally, once the thermomechanical laminae stresses have been calculated, the micromechanical model is used to assess the size of the stresses in the matrix and the fiber. Because of the nature of the titanium-silicon carbide system, the micromechanical model employed here assumes that the interface between the fiber and matrix is weak. Once the thermally induced residual stresses are overcome by laminae stresses, the fiber and matrix will separate. This model was described in the Phase 1 final report. It is three-dimensional in nature and incorporates all thermomechanical loads, both due to processing and engineering environment. Within this software system it is applied under a linear elastic assumption to predict the onset of both interface separation and matrix yielding. The relationship between these programs is shown in Figure 1.

Examples of Results

A series of examples are now offered illustrating the results of this linked set of models as well as some of the physical aspects of the problem in general. First, consider as a reference the composite system made of Ti-6Al-4V and SCS-6 silicon carbide fibers. The fiber and matrix properties for these materials are listed in Table 1. The relevant composite geometric parameters assumed for this example are a fiber volume content of 35% and a fiber spacing ratio (R), defined in Figure 2, of 1.0. Since this initial example is for unidirectional material, a two-laminae composite is considered with both layers oriented in the direction of the x-axis

SYSTEM FLOW CHART

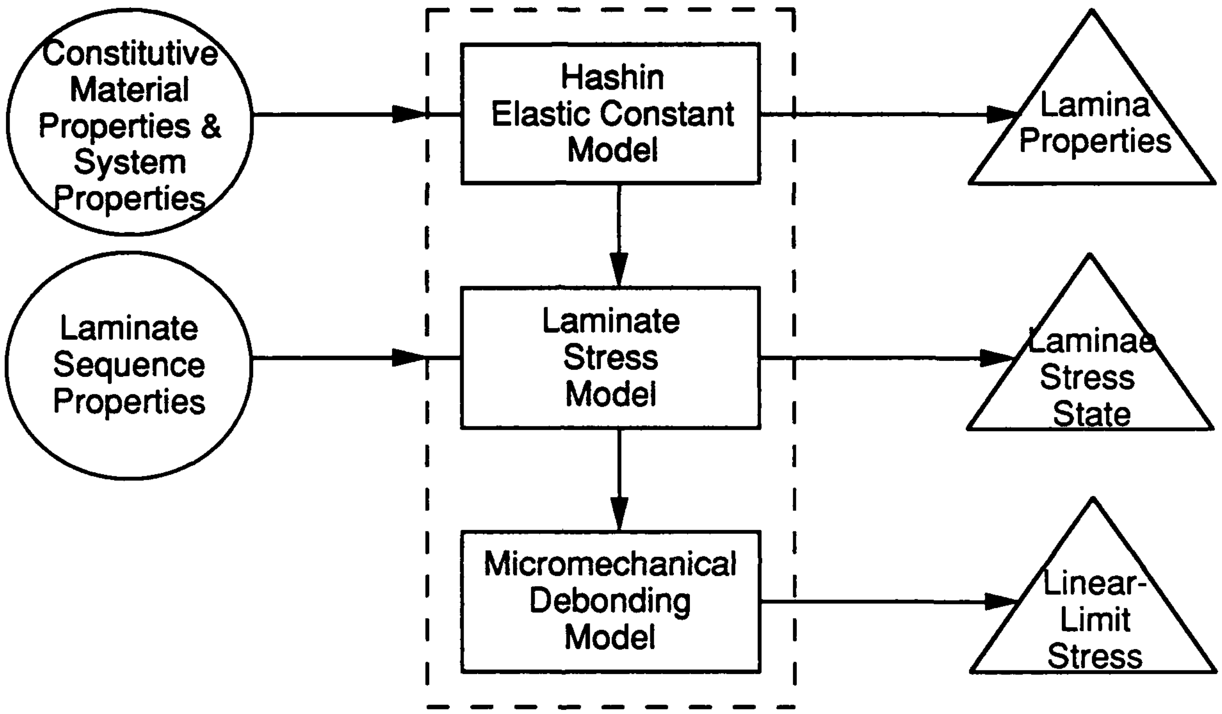


FIGURE 1

Table 1

Matrix Properties (Ti-6Al-4V)	**Fiber Properties (SCS-6 Silicon Carbide)**
E = 113,750.0 MPa	E = 413,650 MPa
ν = 0.3	ν = 0.3
$\alpha = 10.4 \times 10^{-6}$ $(°C)^{-1}$	$\alpha = 4.6 \times 10^{-6}$ $(°C)^{-1}$
σ_{yield} = 1,000.0 MPa	

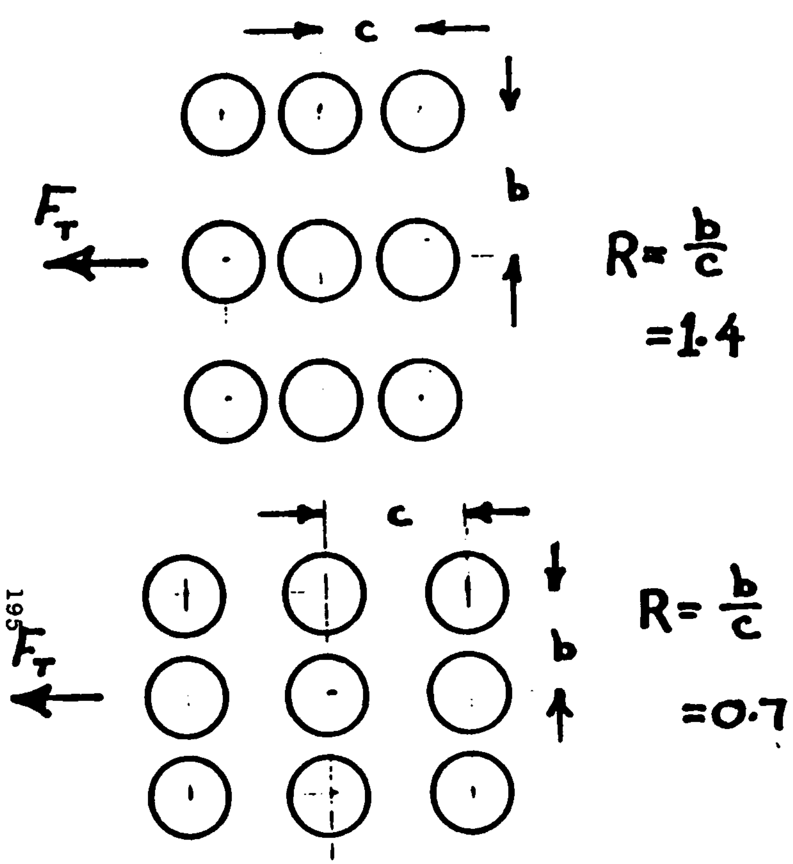
Composite System Properties

$\nu_f = 0.35$

$R = 1.0$

$T_{\text{Ref}} = 900.0°C$

Lay Up: Unidirectional



ASPECT RATIO

DEFINITION

$$R \cong \left(\frac{t}{nr}\right)^2 \frac{v_f}{\pi}$$

$t \triangleq$ specimen thickness

$r \triangleq$ fiber radius

$n \triangleq$ number of fibers through thickness

$v_f \triangleq$ fiber volume fraction

FIGURE 2

(0-degrees).

Since the composite is not cross-plied, there are no average lamina stresses induced in the composite during cooling. There are, however, residual stresses induced in the composite at a micromechanical level due to the difference in coefficients of expansion for fiber and matrix. These micromechanical stresses are summarized in Table 2. The locations of the micromechanical stresses listed in Table 2 are identified in Figure 3. Attempts to measure residual stresses in the matrices of these composites has been attempted by several people. These measurements are average residual stresses representative of distances of the order of 10 - 20 fiber diameters. Experiments (under other funding) carried out on a 35% volume fraction composite of SCS-6 and Ti-6Al-4V measure the average residual stresses perpendicular to the fiber direction to be 175 - 200 MPa; finite element calculations based upon a regular square array of fibers predicts an average transverse residual stress of approximately 250 MPa (time-independent calculations); the simple model offered here predicts an average residual matrix stress in the transverse direction of 310 MPa. In the direction of the fibers, experimental measurements indicate average matrix residual stresses of the order of 275 - 345 MPa; detailed numerical analysis predicts average stresses of 375 MPa in the matrix parallel to the fiber; the simple model used here predicts stresses of 480 MPa. It should be emphasized that the experimental measurements in this case are difficult and their repeatability and accuracy has not been completely established. In light of this fact, the performance of the simple model is deemed to be reasonable. It is also worth noting that the micromechanical model response summarized in Table 2 does not predict yielding on its scale for this composite system. This is consistent with with more detailed analyses.

Table 2

Micromechanical Residual Stresses After Consolidation

(Constituent and Composite System Properties from Table 1)

Fiber

σ_{xx} = -897 MPa
σ_{yy} = -310 MPa
σ_{zz} = -310 MPa
τ_{yx} = 0.0
σ_{eff} = 587 MPa

Matrix

Element A

σ_{xx} = 483 MPa
σ_{yy} = 450 MPa
σ_{zz} = -115 MPa
τ_{yx} = 0.0
σ_{eff} = 587 MPa

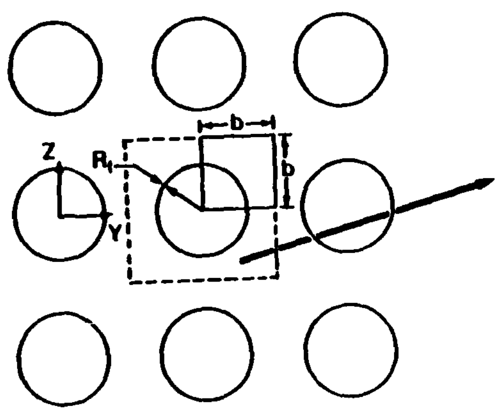
Element B

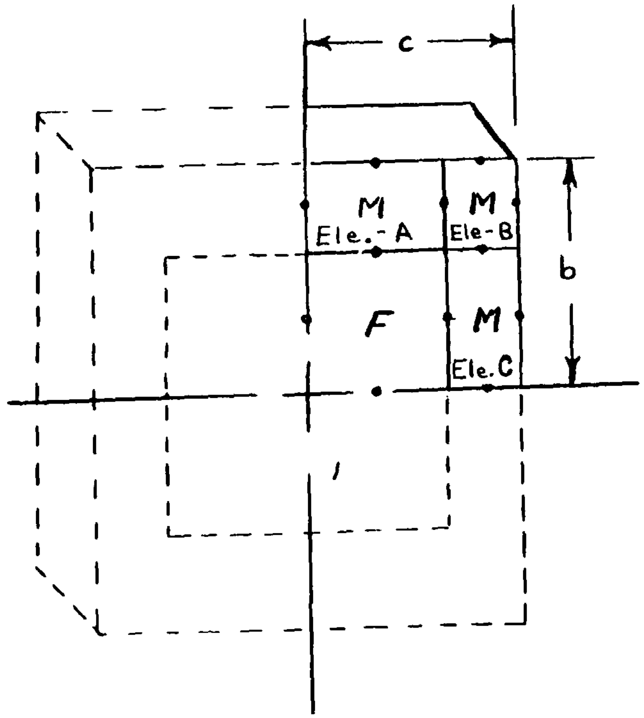
σ_{xx} = 483 MPa
σ_{yy} = 167 MPa
σ_{zz} = 167 MPa
τ_{yx} = 0.0
σ_{eff} = 316 MPa

Element C

σ_{xx} = 483 MPa
σ_{yy} = -115 MPa
σ_{zz} = 450 MPa
τ_{yx} = 0.0
σ_{eff} = 582 MPa

INFINITE, SQUARE ARRAY
OF FIBERS IN A MATRIX





F = Fiber
M = Matrix

FIGURE 3

(7)

For comparison, consider the composite in which the matrix is a titanium aluminide with properties defined in Table 3. For a unidirectional composite with a 35% fiber volume fraction of SCS-6 fibers, the residual stresses predicted by the micromechanical model are listed in Table 4. The first important thing to note is that the matrix represented by elements A and C has almost reached yield. The simple model predicts an average matrix stress in the fiber direction of 420 MPa and an average stress in the direction perpendicular to the fibers of 260 MPa. In comparison, experimental measurements made with respect to this material indicate stresses of 345 - 500 MPa and 165 - 280 MPa in the fiber and transverse direction respectively and detailed finite element models predict 490 - 550 MPa and 190 - 290 MPa.

Now consider a laminated composite made with an SCS-6/Ti-6-4 system and a 35% fiber volume fraction. Consider three laminates defined as (0/90/)s, (60/-60/0)s and (90/45/-45/0)s. All of these laminates must be cooled from 900°C processing temperature, and the composites constituent properties are reported in Table 1. It is of interest, that all three of these laminates develop the same average laminate stresses during cooling. The average lamina longitudinal stress in each case is -127.0 MPa and the average transverse stress is +127 MPa, reported in Table 5. As a result of these average laminate stresses, the micromechanical residual stresses are different in the laminated panels than in the unidirectional panels. The micromechanical residual stresses are now as listed in Table 5. As can be seen, the average transverse tensile force developed in the laminates reduces the radial compression on the fiber in the plane of the lamina. However, the mismatch in coefficients is not sufficient to induce fiber matrix separation during cooling. In addition, the micromechanical stress in the matrix between the individual laminae and in a direction perpendicular to the fibers in the plane of the lamina is increased. However, this increase is not sufficient to

Table 3

Matrix Properties (Ti-14Al-21Nb)	**Fiber Properties (SCS-6 Silicon Carbide)**
E = 97,000 MPa	E = 413,650 MPa
$\nu = 0.3$	$\nu = 0.3$
$\alpha = 12.5 \times 10^{-6}$ (°C)$^{-1}$	$\alpha = 4.6 \times 10^{-6}$ (°C)$^{-1}$
σ_{yield} = 550 MPa	

Composite System Properties

$\nu_f = 0.35$

$R = 1.0$

T_{Ref} = 950.0°C

Lay Up: Unidirectional

Table 4

Micromechanical Residual Stresses After Consolidation

(Constituent and Composite System Properties from Table 3)

Fiber

σ_{xx} = -782 MPa
σ_{yy} = -261 MPa
σ_{zz} = -261 MPa
τ_{yx} = 0.0
σ_{eff} = 520 MPa

Matrix

Element A

σ_{xx} = 420 MPa
σ_{yy} = 378 MPa
σ_{zz} = -97 MPa
τ_{yx} = 0.0
σ_{eff} = 498 MPa

Element B

σ_{xx} = 420 MPa
σ_{yy} = 140 MPa
σ_{zz} = 140 MPa
τ_{yx} = 0.0
σ_{eff} = 280 MPa

Element C

σ_{xx} = 420 MPa
σ_{yy} = -97 MPa
σ_{zz} = 378 MPa
τ_{yx} = 0.0
σ_{eff} = 498 MPa

Table 5

Laminae Stresses After Cooling from 900°C

Matrix and Fiber Properties – Table 1
Lay Up: [90/0]s, [60/-60/0]s, [90/45/-45/0]s

Lamina Residual Stresses (All Laminae)

σ_{LL} = -127 MPa
σ_{TT} = 127 MPa
τ_{LT} = 0.0

Micromechanical Residual Stress State (All Laminae)

Fiber

σ_{xx} = -1167 MPa
σ_{yy} = -151 MPa
σ_{zz} = -321 MPa
τ_{yx} = 0.0
σ_{eff} = 942 MPa

Matrix

Element A

σ_{xx} = 416 MPa
σ_{yy} = 532 MPa
σ_{zz} = -130 MPa
τ_{yx} = 0.0
σ_{eff} = 615 MPa

Element B

σ_{xx} = 431 MPa
σ_{yy} = 258 MPa
σ_{zz} = 192 MPa
τ_{yx} = 0.0
σ_{eff} = 214 MPa

Element C

σ_{xx} = 447 MPa
σ_{yy} = 37 MPa
σ_{zz} = 466 MPa
τ_{yx} = 0.0
σ_{eff} = 419 MPa

create an average effective stress in this area above the matrix yield stress.

Now consider a mechanical load applied to two different laminates. First, a transverse tensile load is applied to a simple unidirectional laminate of 35% fiber volume fraction SCS-6/Ti-6-4 composite. The micromechanical model predicts that the first damage state to occur in this loading is separation of the fiber and the matrix. The micromechanical model predicts that this damage will occur at 250.0 MPa (36.0 ksi). Next consider a cross-plied laminate defined as (90/0)s. When this laminate is loaded unidirectionally, the lamina with fibers oriented transverse to the loading direction suffers interface separation at an applied laminate stress level of 142 MPa (21 ksi). If the effects of laminate induced residual stresses had been ignored and transverse experimental transverse data used in conjunction with a mechanically loaded laminate analysis, a limit stress of 275 MPa (40.0 ksi) would have been predicted. This difference is due to residual stresses induced during cross-ply fabrication. Figure 4 illustrates the linear elastic stress-strain response in the orthogonal directions of the unidirectional panel and the (90/0)s cross-plied panel as predicted by this model. As can be seen from the figure, the laminating process does not have a significant effect upon stiffness. However, the model predicts early nonlinearity in both orthogonal directions as a result of fiber-matrix interface separation.

CONCLUSIONS

Failure criteria for metal matrix composites are related to manufacturing and processing through the residual stresses which are induced during cooling from process temperature. This is a significant result in that damage observed experimentally on unidirectional composite panels is affected by the sequence of laminated panels. The differences are due to additional residual stresses incurred when the individual lamina are cross-plied and consolidated.

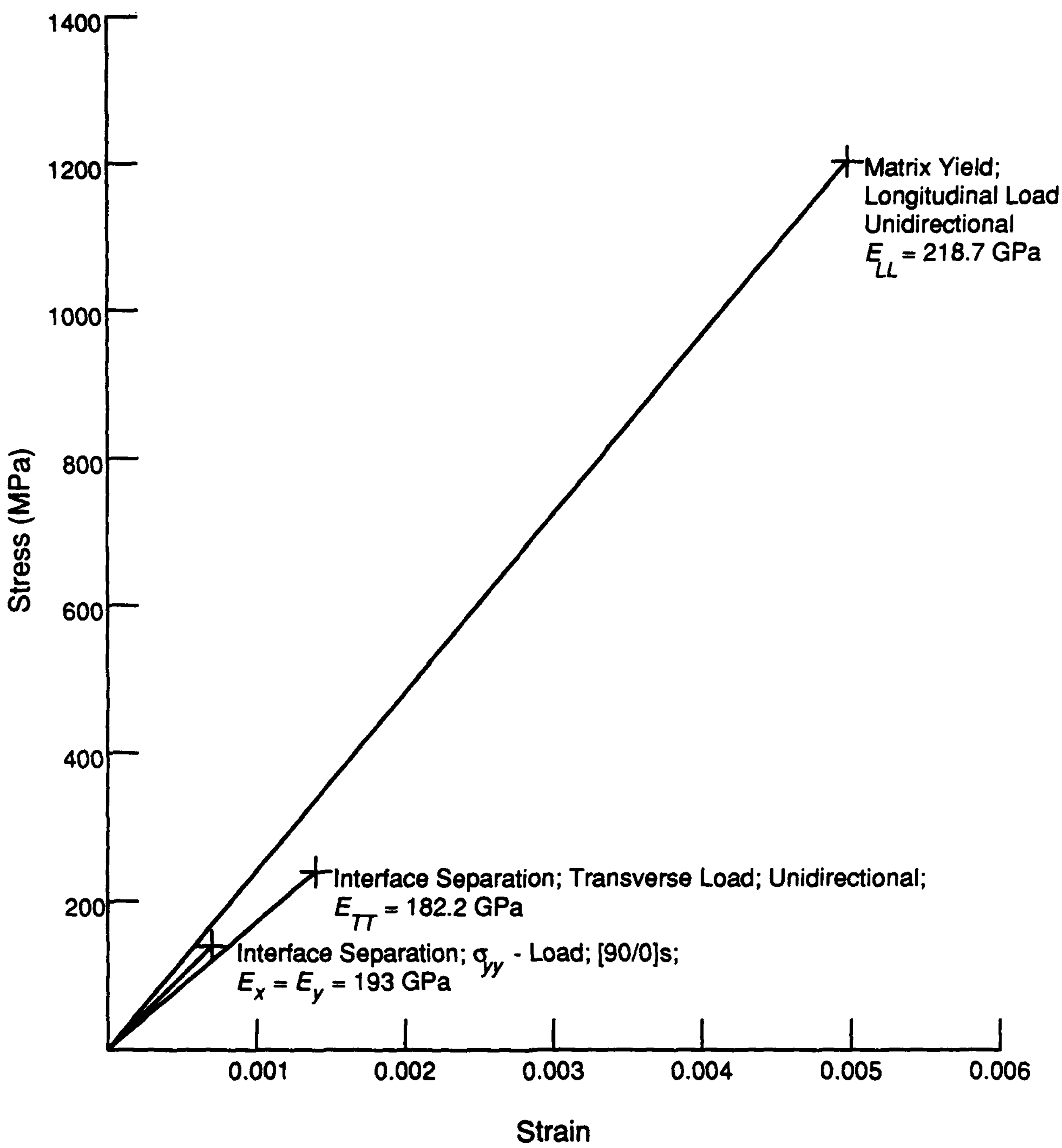


FIGURE 4

Predicted First Nonlinearity in Unidirectional Tensile Test

The software discussed here provides a first order vehicle to assess and predict these effects. Its simplicity makes it very easy to incorporate with optimization tools under development within this overall program. The simplicity of the models results in some sacrifice with respect to accuracy. However, once critical parameter ranges have been identified, more accurate models can be applied to provide quantitative predictions.

The tools discussed here could also be expanded to treat problems of dimensional stability in MMC composites.

RECOMMENDATIONS

The models and predictions made here provide a very useful approach for concurrently engineering new materials and components. The tools need to be verified with respect to their accuracy in predicting physical events – specifically failure events. An approach to accomplishing this task has been proposed under Phase 3 DICE funding and is underway. That task could be made more effective if carried out in close collaboration with material and process related personnel.

PUBLICATIONS

None.

HARDWARE

None.

Task 4.4.5.2 ICPD Material Properties GEAE

Objectives:

A materials properties data base for titanium aluminide composite materials made by the ICPD process is being generated for use in the micromechanical models and design rules being developed under this contract. The effects of processing parameters which influence these properties are being isolated and eliminated if found to have deleterious effects. The importance of matrix ductility and toughness to composite mechanical properties is being determined by a systematic screening of candidate advanced alpha two alloys which may eventually replace Ti-14Al-21Nb. The effect of consolidation temperature on matrix microstructure and composite properties has also been examined.

Approach:

Phase II of this task covered the machining and testing of mechanical test specimens cut from a 6" x 15" Ti-14Al-21Nb/SCS-6 composite panel which was sprayed and canned at CR&DC and HIPed at Howmet. This 4-ply panel was designated panel E. The drums were wound to 112 filaments per inch. The starting powder chemistry (-80 +140 mesh) was 15.5 % aluminum and 20.4 niobium, balance titanium. This material was plasma sprayed with 3 % H_2 to result in a nominal matrix composition of 14 % aluminum and 21 % niobium. Fibers were extracted from the plasma sprayed foils in order to measure fiber tensile strengths. The panel was laid up with four unidirectional monotapes and three neat foils, one in the middle and one on each side. This procedure resulted in a panel with a relatively low volume fraction (approximately 22 %). The panel was HIPed at 1832°F (1000°C) for 3 hours at 15 ksi (103 MPa). The panel was inspected by c-scan and machined into specimen blanks. Chemical analyses for interstitial levels were made. Three blanks were returned to CR&DC for as-HIPed fiber extraction. Some of the specimen blanks were given a simulated secondary bonding heat treatment of 3 hours at 1850°F, and the molybdenum rich layer of some of the blanks was removed either by diamond grinding or acid cleaning in a solution of 56% H_2O, 4% HF and 40% HNO_3.

A second panel was manufactured during Phase II in order to use for mechanical testing to correlate with the Phase II MMC disk (Task 4.5.4.2). The Ti-14Al-21Nb monotape and foil were sprayed at Lynn with the same fiber spacing (110 filaments/inch) as the disk and a deposition rate calculated to produce the same volume fraction of SCS-6 fibers as found in the disk. The tapes were then HIPed into a 4" x 5" panel at CR&DC at 1832°F (1000°C) and 15 ksi. After cutting the panel into blanks, portions of it were given a simulated secondary bonding cycle, and longitudinal, transverse and through-thickness tensile specimens were fabricated. Interstitial contents were also determined. Extracted fiber strengths and percent unbroken fibers were determined from two blanks in the as-HIPed and as-heat treated conditions.

Four additional panels were received from CR&DC at the very end of Phase 2. Although the panels were machined into specimens, there was not enough time for testing to be completed during Phase 2. This work will proceed during Phase 3.

In addition, a first attempt was made to screen advanced alpha two matrix alloys in order to identify an improved matrix to replace the Ti-14Al-21Nb currently being used. (This effort was performed in conjunction with effort under Task 4.5.4.2). It is felt that eliminating the beta depleted region around the SCS-6 fibers in Ti-14Al-21Nb matrix composites would lead to reduced matrix microcracking. Three alloys were chosen: Ti-13Al-31Nb, Ti-13Al-15Nb-4Mo-2V-7Ta and Ti-13Al-11Nb-4Mo-2V-7Ta (wt%). Ti-14Al-21Nb was used as a baseline. These alloys were plasma sprayed in both neat and composite (SCS-6 fibers wound at 115 filaments/inch) form. The baseline panels were sprayed with 3 % H_2 to result in a nominal matrix composition of 14 % aluminum and 21 % niobium and consolidated in the vacuum hot press at Lynn at 1832°F for 3 hours at 10 ksi. All of the advanced alpha two alloys were VHPed at 1950°F for 3 hours at 10 ksi, except the Ti-13-11-4-2-7 neat panel, which was HIPed at CR&DC at 1832°F for 3 hours at 15 ksi. Interstitial contents during processing were tracked, along with percent unbroken fibers and extracted fiber strength. Each matrix alloy was subjected to four different heat treatments in an attempt to reproduce wrought microstructures, after which tensile and toughness tests were performed at room temperature. After evaluating these results on the neat material, 2 heat treatments were chosen for the composite materials, and room temperature tensile tests were performed on these materials.

Fiber push through tests at room temperature and elevated temperatures were performed on titanium aluminide/SCS-6 composites at CR&DC in order to determine, qualitively at least, differences in interfacial bond strengths, so that composite properties could be related to interfacial characteristics.

An effort was initiated at CR&DC to study how various hot isostatic pressing (HIP) consolidation times and temperatures affect the matrix microstructure, reaction zone layer and tensile properties of Ti-14Al-21Nb/SCS-6 composite materials. The composite panels were fabricated by plasma spraying Ti-14Al-21Nb (-80 +140) powder onto a drum wound with fibers having a spacing of 112 fibers/inch. Various hydrogen contents of the plasma gases were used. Interstitial analyses were done on the tapes. The amount of fiber breakage and the average fiber strength after fabrication were determined. The plates were cut into 0 and 90 degree blanks for tensile testing at room temperature, and the fracture surfaces were examined by scanning electron microscopy (SEM).

Technical Results:

Three blanks taken from different locations in panel E (CR&DC notation 1174) showed extracted fiber strengths (as HIPed) of 508 $\pm$ 91 ksi (E-12), 526 $\pm$ 104 ksi (E-25), and 570 $\pm$ 45 ksi (E-30), indicating satisfactory fiber strength, with minimal damage from the plasma spray process. A c-scan and metallography revealed poor fiber spacing, however, and interstitial contents were high (5140 ppm O, 582 H, 105 ppm N). As expected, results for the room temperature tensile tests showed very poor room temperature tensile properties (Table 4.4.5.2-1). This high oxygen content was probably due to a leak in the plasma spray system at CR&DC during spraying of the large drum, and would be expected to lead to matrix embrittlement and very low ductility, which has been shown to reduce the overall matrix strength. Low matrix strengths lead to low composite strengths, based on a rule-of-mixtures calculation. This was indeed observed in this panel, where all longitudinal test strengths were around 90 ksi. Surface removal through diamond grinding or acid cleaning and secondary heat treatments were found to increase the tensile strength a marginal amount (approximately 10%). However, interrupted tensile tests were inconclusive in determining if cracks initiated from the molybdenum rich surface layer or internally.

Room temperature tensile tests were also performed on specimens which had been impacted at 1200°F during Phase I. Results are shown in Table 4.4.5.2-2. Longitudinal tensile specimens taken from the same panel (B) exhibited an average room temperature tensile strength of 138.7 ksi (2 specimens), so even at the higher impact energy, the longitudinal samples retained about 60 percent of their strength. Strength retention in the transverse specimens appeared to be near 100 percent for both impact energies (although a larger specimen was used) since a specimen taken from panel D exhibited a non-impacted room temperature tensile strength of 46.8 ksi.

SPECIMEN NUMBER	TEMPERATURE $^{o}F(^{o}C)$	TEST ORIENTATION	UTS KSI(MPa)	E MSI(GPa)	e_f (%)	VOLUME FRACTION	FAILURE LOCATION
E-13	70(21)	0[4]	87.7(605)	21.8(150)	0.45	0.22	radius
E-15(1,3)	70(21)	0[4]	(2)	23.5(162)			
E-16(1,3)	70(21)	0[4]	(2)	23.6(163)			
E-17(1,3)	70(21)	0[4]	96.0(662)	23.9(165)	0.42	0.23	gage
E-20(3)	70(21)	0[4]	(2)	23.5(162)			
E-21(3)	70(21)	0[4]	(2)	22.7(157)			
E-22(3)	70(21)	0[4]	106.3(733)	22.0(152)	0.50	0.21	gage
E-23	70(21)	0[4]	84.3(581)	22.4(154)	0.40	0.21	radius
E-31	70(21)	0[4]	88.6(611)	22.3(154)	0.39	0.21	radius
E-35(3,4)	70(21)	0[4]	(2)	23.3(161)			
E-36(3,4)	70(21)	0[4]	(2)	23.1(159)			
E-37(3,4)	70(21)	0[4]	100.9(696)	23.2(160)	>0.43	0.24	gage

(1) Specimen had diamond ground surface.
(2) Specimen was not tested to failure.
(3) Specimen heat treated at 1850°F for 3 hours.
(4) Specimen had acid cleaned surface.

Table 4.4.5.2-1 Tensile Properties of Ti-14-21/SCS-6 Unidirectional Composite HIPed at 1832°F (1000°C) for 3 hours at 15 ksi (103 MPa)

SPECIMEN NUMBER	TEST ORIENTATION	IMPACT TEMPERATURE °F(°C)	SPECIMEN WIDTH inches	ENERGY ft-lbs	External visual appearance
B-2	0[4]	1200(649)	0.400	1.08	Large indent
	RT tensile strength after impact 83.7 ksi (577 MPa)				
B-5	0[4]	1200(649)	0.399	0.502	Slight indent
	RT tensile strength after impact 115.8 ksi (798 MPa)				
C-1	90[4]	1200(649)	0.900	0.502	Slight indent
	RT tensile strength after impact 46.9 ksi (323 MPa)				
C-2	90[4]	1200(649)	0.899	1.07	Large indent
	RT tensile strength after impact 41.2 ksi (284 MPa)				

Table 4.4.5.2-2 Ballistic impact test results with subsequent tensile strengths for Ti-14Al-21Nb/SCS-6 unidirectional composite HIPed at 1832°F (1000°C) for 3 hours at 15 ksi (103 MPa)

Results from the room temperature tensile tests which were performed on the Ti-14Al-21Nb panel to be used to correlate with the spin test results are shown in Table 4.4.5.2-3. (The through thickness tensile tests have not been performed as yet.) The as received fiber strength was 456.4 $\pm$ 50.5 ksi (7 tests). After spraying, the extracted fiber strength was 628.8 $\pm$ 29.7 ksi (10 tests). These numbers obviously reflect a deficiency in the method of calculating fiber strength or in the sample group size or location tested, since the as-sprayed fiber strengths should be equal to or lower than the as-received values. Fiber extractions were also performed on blanks taken from the panel, in the as-HIPed and as-heat treated (1850°F, 3 hours) conditions. Results showed 66 percent unbroken fibers in the as-HIPed blank (GL-1) and 80% unbroken fibers in the HIPed plus heat treated specimen (GL-6). The average extracted fiber strength for GL-1 was 473.3 $\pm$ 239.4 ksi; for GL-6 it was 565.7 $\pm$ 108.6 ksi (10 tests each). These strengths indicate a loss of strength over the as sprayed condition of 10-20 %. It seems odd that the percent unbroken fibers was higher in the heat treated blank than in the as-HIPed specimen, and that the extracted fiber strength was higher in the as heat treated specimen. Interstitial contents are shown in Table 4.4.5.2-4, and show extremely high oxygen pick ups during the spraying process. These interstitial values are probably too high to generate good tensile properties. The longitudinal tensile strengths measured from this panel were low, and did not meet the rule of mixtures criterion. However, the transverse tensile strength was quite high, probably because of the low volume fraction.

In an effort to identify an alternative alpha two matrix alloy and to determine the importance of matrix ductility and toughness on composite properties, the potential matrix alloys and baseline Ti-14Al-21Nb were compared. High interstitial contents, especially in the Ti-13Al-15Nb-4Mo-2V-7Ta and Ti-13Al-11Nb-4Mo-2V-7Ta alloys (see Table 4.4.5.2-5) may have decreased the ductilities of the neat panels, while their room temperature tensile strengths were high (see Table 4.4.5.2-6). The selected heat treatments reproduced the wrought microstructures in the plasma sprayed matrix, at least at the level of optical microscopy, but these microstructures were not found to have the same toughnesses and ductilities as their wrought counterparts, even though their strengths were high. One possible explanation for this is that the high interstitial contents have destroyed the matrices' inherent ductility. The results from chemical analyses of the neat material are shown in Table 4.4.5.2-7.

Tensile results for composites made from these matrices are shown in Table 4.4.5.2-8. As can be seen, none of the alternative matrices showed substantial improvement over Ti-14Al-21Nb in ductility or strength, and the two advanced alpha two alloys tested exhibited a substantial amount of matrix cracking. Significantly, the composite tensile strength of Ti-14Al-21Nb was higher than that of Ti-13Al-31Nb, even though in neat form the Ti-13Al-31Nb was stronger. Ti-14Al-21Nb had greater ductility in the neat form, however, indicating that matrix ductility (or toughness) is more important to composite tensile properties than is matrix strength.

SPECIMEN NUMBER	TEST ORIENTATION	UTS KSI(MPa)	E MSI(GPa)	e_f (%)	VOLUME FRACTION	FAILURE LOCATION
GL-2*	4[0]	122.8(847)	20.5(141)	0.87	13.9	gage
GL-4*	4[0]	135.5(934)	20.5(141)	1.05**	14.1	radius
GT-1*	4[90]	54.7(377)	18.2(125)	0.36		gage
GL-3	4[0]	124.0(855)	20.6(142)	0.84**	13.9	radius
GL-5	4[0]	117.2(808)	20.1(139)	0.79	14.2	gage

* Given simulated secondary bonding cycle of 3 hours at 1850°F (1000°C).

** Apparent elongation due to specimen failure outside extensometer not included in average.

Table 4.4.5.2-3 Room temperature tensile results for Ti-14Al-21Nb panel used to correlate with spin test results

		Powder	As-sprayed
LPS 284	O	860	2072
	N	87	146
	C	100	288
	H	10	43
LPS 285	O	860	1480
	N	87	120
	C	100	226
	H	10	59

Table 4.4.5.2-4 Interstitial contents (in ppm) of powder and as-sprayed monotape and foil for Ti-14Al-21Nb panel used to correlate with spin test results

		Powder	Neat Panels
Ti-14Al-21Nb - VHP 75	O	860	1035
	N	87	65
	C	100	503
	H	10	11
Ti-13Al-31Nb - VHP 126	O	870	1100
	N	60	120
	C	27	210
	H	140	9
Ti-13Al-11Nb-4Mo-2V-7Ta - RF 993	O	1090	1450
	N	60	130
	C	36	280
	H	220	120
Ti-13Al-15Nb-4Mo-2V-7Ta - VHP 129	O	1600	2082
	N	116	244
	C	15	396
	H	156	3

Table 4.4.5.2-5 Interstitial contents of powders and neat panels for advanced alpha two alloys

SPECIMEN NUMBER	UTS KSI(MPa)	E MSI(GPa)	e_f (%)	K_L (in)
Ti-14Al-21Nb				
75-AR	87.0(600)	14.1(97)	3.22	18.1,17.6
75-D1	80.6(556)	14.6(101)	0.69	11.8
75-E1	91.1(628)	15.1(104)	>2*	16.8
75-F1	87.7(605)	14.2(98)	>2*	20.9
75-S2	103.3(712)	14.2(98)	1.88	11.3
Ti-13Al-31Nb				
126-AR	117.8(812)	17.0(117)	0.94	18.8,15.1
126-A1	114.4(789)	17.8(123)	1.50	11.2
126-S1	107.7(743)	18.2(125)	0.60	11.8
126-B1	110.2(760)	17.0(117)	0.66	11.0
126-C1	110.9(765)	17.0(117)	0.71	11.9
Ti-13Al-11Nb-4Mo-2V-7Ta				
933-AR	48.7(336)	17.5(121)	0.28	12.4,12.6
933-A1	109.8(757)	16.6(114)	0.68	8.55
933-S1	98.7(681)	16.4(113)	0.60	8.92
933-B1	97.2(670)	17.3(119)	0.57	8.65
933-C1	82.8(571)	16.9(117)	0.49	9.51
Ti-13Al-15Nb-4Mo-2V-7Ta				
129-AR	83.9(578)	16.4(113)	0.53	11.0,10.6
129-A1	116.1(800)	17.5(121)	~0.5*	7.58
129-S1	74.1(511)	18.1(125)	~0.5*	7.86
129-B1	130.3(898)	17.8(123)	~0.5*	7.89
126-C1	92.7(639)	17.6(121)	0.53	7.24

* Extensive noise from matrix cracking made exact strain determination impossible.

A1 # 2150°F/6 min, He quench, 1600°F/1 hr, He quench, 1300°F/8 hr
B1 # 1950°F/1 hr, cool 10°F/min, 1300°F/8hr
C1 # 1950°F/1 hr, He quench, 1600°F/1 hr, He quench, 1300°F/8 hr
D1 # 1850°F/1 hr, vacuum cool
E1 # 1950°F/1 hr, vacuum cool, 1400°F/8 hr
F1 # 2050°F/15 min, vacuum cool, 1400°F/8 hr
S1 + 2150°F/6 min, salt quench, 1500°F/1 hr, air cool, 1300°F/8 hr
S2 + 2125°F/15 min, salt quench, 1400°F/8 hr

Note: # denotes vacuum heat treat
+ denotes quartz encapsulation

Table 4.4.5.2-6 Tensile properties and toughness values for neat advanced alpha two alloys

	Ti	Al	Nb	Mo	V	Ta
Ti-14-21 VHP 75	64.7	14.2	21.0	<0.1		
Ti-13-31 VHP 126	56.2	13.7	28.4	0.43		
Ti-13-11-4-2-7 RF 993	61.7	14.0	12.7	3.75	2.25	6.32
Ti-13-15-4-2-7 VHP 129	58.6	13.8	14.6	3.86	2.01	7.08

Table 4.4.5.2-7 Chemical analyses of neat advanced alpha two alloys

SPECIMEN NUMBER	UTS KSI(MPa)	E MSI(GPa)	e_f (%)	VOLUME FRACTION	FAILURE LOCATION
Ti-14Al-21Nb					
142-S2	157.4(1085)	26.1(180)	0.90	27.4	gage
142-S2	174.6(1204)	25.7(177)	1.03	30.9	gage
142-F1	182.7(1260)	28.1(194)	1.05	29.2	gage
142-F1	180.7(1246)	27.3(188)	0.96	29.7	gage
142-AR	177.3(1222)	27.7(191)	1.00	24.8	gage
Ti-13Al-31Nb					
132-A1	132.9(916)	29.6(204)	0.54	30.3	gage
132-A1	114.9(792)	29.2(201)	0.45	30.7	gage
132-S1	135.0(931)	28.8(199)	0.58	29.5	radius
132-S1	120.1(828)	30.3(209)	0.46	30.6	gage
132-AR	99.9(689)	30.0(207)	0.36	28.7	radius
Ti-13Al-11Nb-4Mo-2V-7Ta					
84-C1	117.7(812)	29.7(205)	*	28.4	gage
84-A1	123.1(849)	29.6(204)	*	30.1	gage
84-AR(1200°F)	136.9(944)	25.7(177)		28.1	radius
Ti-13Al-15Nb-4Mo-2V-7Ta					
133-A1	110.3(760)	29.9(206)	*	30.0	gage
133-B1	99.1(683)	29.8(205)	*	30.0	radius
133-A1	99.5(686)	29.6(204)	*	30.7	gage
133-B1	92.1(635)	30.5(210)	*	30.6	radius
133-AR	118.9(820)	27.5(190)	*	29.5	gage

* Could not detect due to extensive noise from matrix cracking.

Table 4.4.5.2-8 Tensile properties of advanced alpha two composites. Heat treatments same as in Table 4.4.5.2-5

In the HIP consolidation study performed at CR&DC, HIP temperatures ranged from 1650 to 1950°F (900 to 1065°C), and time at both temperature and pressure ranged from 0.5 to 3 hours. These are the times and temperatures at peak. However, it should be noted that the time to reach the HIP temperature, the time to pressurize the chamber to 15 ksi and the total time at the HIP temperature varied. Average interstitial analyses for the tapes fabricated at CR&DC for the HIP consolidation study are given in Table 4.4.5.2 - 9.

There was a trend of decreasing beta phase fraction with increasing time at temperature for the panels HIPed at temperatures between 1650 and 1830°F (900-1000°C). After HIP at 1950°F (1065°C), the matrix consisted of transformed beta phase. There was no obvious correlation between the reaction zone thickness and either the total time at the HIP temperature, or the total time at which the panel was at both full temperature and pressure. The extent of the beta depleted region increased with increasing HIP temperature and increasing time at temperature.

The room temperature tensile properties obtained from these panels are shown in Table 4.4.5.2 - 10. Because of high interstitial contents and poor fiber winding, the tensile properties from panels RF1173-3 were impossible to correlate with any known parameters. However, the longitudinal properties of the lower interstitial panels were affected by the HIP temperature, the fiber strength after HIP, and the fiber volume fraction. For example, the longitudinal tensile strength of the panels appeared to decrease as the HIP temperature increased, but was most strongly influenced by the as-HIP fiber strength. There was also a substantial effect of fiber strength on the longitudinal tensile strength.

Results from the fiber push through tests are shown in Table 4.4.5.2 - 11. After initial indentation at room temperature, some of the samples were inverted so that additional pushes could be performed. This was done in an attempt to separate the chemical bond strength of the interface from the frictional bond strength. As expected, results from second pushes were significantly less than the first push. A great deal of non-uniformity was observed in some samples, where one particular row of fibers would show much greater or lower values than the rest of the sample. There was good agreement between tests done at two different times for X20A-07, but not for 284-HT. A possible explanation for this is that the higher values for sample 284-HT were taken from a thicker sample. Figure 4.4.5.2 - 1 shows a pushed through fiber. These results show that it is possible to separate the chemical bond strength of an interface from the frictional bond strength by the fiber push through method,

RF No.	Oxygen	Nitrogen	Total
1171	2189 ± 369 ppm	490 ± 56 ppm	2679 ppm
1172	5395 ± 447	125 ± 7	5520
1173	5943 ± 298	188 ± 13	6131
1223	1509 ± 1	896 ± 15	2405
1224	1522 ± 126	386 ± 31	1908
1225	1379 ± 147	616 ± 147	1995
1275	1261 ± 16	456 ± 8	1717
1276	1030 ± 32	232 ± 46	1262
1277	1053 ± 66	275 ± 30	1328
1279	1176 ± 84	226 ± 45	1402
1280	1207 ± 129	425 ± 34	1532

Table 4.4.5.2 - 9 Average interstitial analyses of plasma-sprayed tapes

PANEL NUMBER	HIP CONDITIONS	TEST ORIENTATION	UTS KSI(MPa)	E MSI(GPa)	e_f	VOLUME FRACTION
1171A	1830F(1000C)/3hr	0[4]	121(834)	22.9	0.66	0.22
1171B	" " "	0[4]	139(958)	23.3	0.78	0.22
1171D	" " "	0[4]	122(841)	22.5	0.69	0.22
1172A	1830F(1000C)/3hr	0[4]	105(724)	24.0	0.52	0.20
"	" " "	90[4]	38(262)	16.4	0.26	"
1171B	1830F(1000C)/3hr	0[4]	89(614)	22.4	0.41	0.20
"	" " "	90[4]	40(276)	16.9	0.26	"
1172C	1650F(900C)/3hr	0[4]	86(593)	22.5	0.42	0.20
"	" " "	0[4]	77(531)	21.4	0.37	"
"	" " "	0[4]	88(607)	21.2	0.43	"
"	" " "	90[4]	32(221)	15.1	0.21	"
1173A	1830F(1000C)/3hr	0[4]	108(745)	24.4	0.50	0.22
"	" " "	90[4]	42(290)	17.9	0.26	"
1173B	1830F(1000C)/3hr	0[4]	94(648)	26.2	0.44	0.22
"	" " "	90[4]	31(214)	18.7	0.18	"
1223A	1740F(950C)/2.5hr	0[4]	130(896)	22.2	0.81	0.17
"	" " "	90[4]	67(462)	16.6	0.60	"
1223B	1830F(1000C)/3hr	0[4]	145(1000)	23.3	0.96	0.19
"	" " "	90[4]	53(365)	18.1	0.37	"
1223C	1950F(1065C)/1hr	0[4]	126(869)	21.5	0.82	0.18
"	" " "	90[4]	61(421)	17.5	0.56	"
1223D	1830F(1000C)/1.5hr	0[4]	153(1055)	21.1	1.09	0.18
"	" " "	90[4]	67(462)	17.3	0.63	"
1224A	1950F(1065C)/0.5hr	0[4]	143(986)	22.8	0.86	0.20
"	" " "	90[4]	49(338)	17.2	0.43	"
1224B	1740F(950C)/1.3hr	0[4]	165(1138)	21.3	1.09	0.21
"	" " "	90[4]	59(407)	16.0	0.51	"
1225A	1950F(1065C)/0.5hr	0[4]	107(738)	22.2	0.65	0.16
"	" " "	0[4]	119(820)	20.8	0.80	"
"	" " "	0[4]	130(896)	20.9	0.93	"
"	" " "	90[4]	61(421)	16.3	0.52	"
1225B	1650F(900C)/1.1hr	0[4]	177(1220)	20.4	1.19	0.19
"	" " "	90[4]	57(393)	15.1	0.52	"
1276	1650F(900C)/1.4hr	0[4]	177(1220)	23.4	0.96	0.23
"	" " "	90[4]	60(414)	16.4	0.52	"
1277	1740F(950C)/1.2hr	0[4]	169(1165)	24.7	0.85	0.24
"	" " "	90[4]	54(372)	16.5	0.48	"
1279	1830F(1000C)/0.5hr	0[4]	170(1172)	23.4	0.93	0.22
"	" " "	90[4]	53(365)	16.3	0.53	"
1280	1950F(1065C)/0.5hr	0[4]	157(1082)	23.2	0.88	0.23
"	" " "	90[4]	46(317)	16.3	0.42	"

Table 4.4.5.2 - 10 Room temperature tensile properties

	First push	Second push	Third push
RF 984	12.6 ± 3.5	8.51 ± 1.97	
126	25.5 ± 3.36	16.85 ± 0.81	
L110	13.0 ± 1.97	5.7 ± 0.807	
284-AF*	26.3 ± 3.8		
284-HT	14.8 ± 2.33	7.3 ± 2.8	
284-HT*	21.7 ± 1.9	8.4 ± 0.82	6.43 ± 0.82
X20A-07	21.5 ± 2.2		
X20A-07*	24.57 ± 1.37		(0.053" thick)
X20A-07*	23.8 ± 5.14	8.01 ± 1.35	(0.040" thick)
X20A-2LE	15.9 ± 4.6		

	800°F	1200°F	1600°F
X20A07	12.3 ± 4.0	7.19 ± 1.2	5.0 ± 0.3
X20A-2LE	7.44 ± 0.55	6.33 ± 0.92	3.95 ± 1.2
284-AF*	12.92 ± 1.87	9.79 ± 1.97	6.9 ± 1.05
284-HT	5.56 ± 0.4	4.2 ± 0.22	3.39 ± 0.4

* These tests were done at a different time.

Sample histories:

284-AF	Foil-fiber-foil (FFF) Ti-14Al-21Nb/SCS-6, 8-ply, as HIP
284-HT	284-AF, plus 3 hours in vacuum at 1800°F
X20A-07	FFF Ti-14Al-21Nb/SCS-6, 8-ply, HIPed plus 3 hrs in vacuum at 1800°F
X20A-2LE	X20A-07, plus 100 thermal cycles (70 to 1200°F) and 1 thermal cycle (70 to 1650°F)
RF 984	RSPD Ti-13Al-31Nb/SCS-6, as VHPed
126	RSPD Ti-13Al-11Nb-4Mo-2V-7Ta/SCS-6, as VHPed
110	RSPD Ti 6-2-4-2/SCS-6, as HIPed

Note: This sample was tested at the same time under alternative funding.

Table 4.4.5.2 - 11 Results from fiber push through tests (all results in ksi)

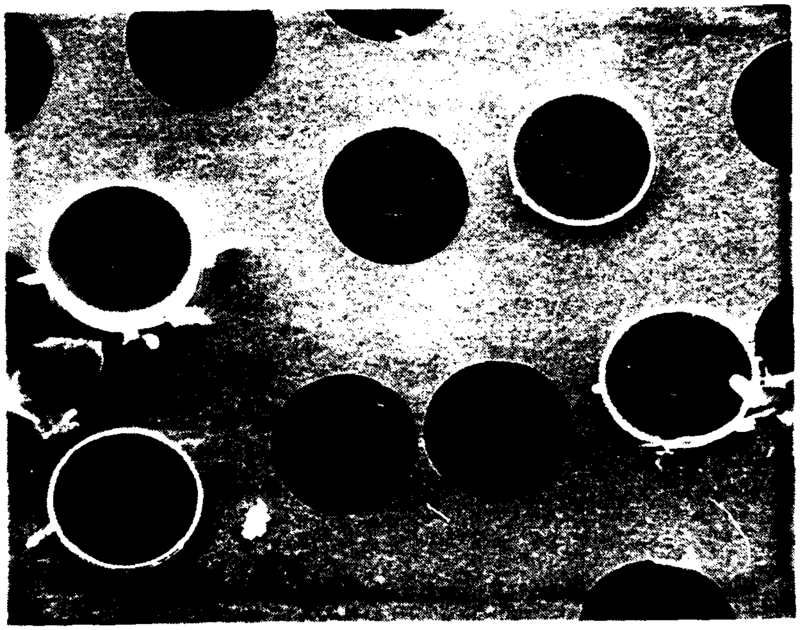


Figure 4.4.5.2-1 SEM micrograph (150X) of pushed through fibers on sample 284-AF, easily identified by white rings around pushed through fibers.

and that it is also possible to track the decrease in interfacial strength as a function of temperature. Raising the temperature of the composite during an elevated temperature test tends to reduce the residual stresses imposed on the fibers by the matrix during cooling from processing temperatures, and thus results in the observed decrease of the interfacial shear stress. A substantial difference in interfacial strength was also noted between a sample which had been thermally cycled (X20A-2LE) and a non-thermally cycled sample (X20A07) of the same material.

Conclusions:

The material received to date to complete this task has had interstitial levels which are too high to enable the establishment of an appropriate data base for the design of the DICE MMC disk. Current mechanical property data on Ti-14Al-21Nb/SCS-6 plasma sprayed composites show insufficient ductility at room temperature and, as expected, do not achieve rule of mixtures strength. Elevated temperature properties, however, are slightly more encouraging, probably because the effects of high interstitials are negated by the effects of elevated temperature testing.

The preferred sites for crack initiation during tensile testing of these materials could not be determined. It is not obvious that cracking initiates in the molybdenum rich surface layer found on many composite panels, although from scanning electron microscopy performed on fractured specimens, it appears that the layer fails in cleavage.

None of the advanced alpha two alloys evaluated in this study exhibited sufficient ductility as composite matrices to warrant replacement of the currently used Ti-14Al-21-Nb. However, the starting interstitial contents of two of the powders (Ti-13Al-11Nb-4Mo-2V-7Ta and Ti-13Al-15Nb-4Mo-2V-7Ta) may have been high enough to destroy the materials' inherent ductility. Plasma sprayed Ti-13Al-31Nb exhibited higher strength than plasma sprayed Ti-14Al-21Nb alloy, but lower ductility. But the Ti-14Al-21Nb alloy exhibited higher composite strength, showing that an alloy's ductility is more important than its strength for composite properties.

Consolidation of high-beta Ti-14Al-21Nb/SCS-6 composites was achieved at times as low as 1.1 hours at 1650°F (900°C), 1.5 hours at 1830°F (1000°C) and 0.5 hours at 1950°F. However, wide variations in matrix microstructure, including the amount, size and distribution of the transformed beta phase regions, were observed. These microstructural differences also led to changes in the fracture mode. Higher HIP temperatures led to decreased yield strengths at room temperature, possibly due in part to changes in the residual stresses in the matrix. The ultimate tensile strengths at room temperature were most strongly influenced by the as-HIP fiber strength, although fiber volume fractions and HIP temperatures may have had an effect.

Recommendations:

The current data base is not large enough to permit disk design calculations or support design rule formulation. The total amount of non-DICE information is probably large enough to determine the relative contributions of fiber volume fractions, strength and percent broken to composite mechanical properties. Although the material received at the end of this phase may help in expanding the data base, there has not been enough good material fabricated during the course of this phase to complete the task. Efforts should continue during Phase 3 to characterize Ti-14Al-21Nb/SCS-6 mechanical properties using improved material. Efforts have been conducted at Textron Specialty Materials Division to qualify them as a supplier of MMC panels, and good quality material is expected to be available from them early in Phase 3.

In order to promote more homogeneous microstuctures and possibly increase matrix ductility, it is recommended that the molybdenum rich surface layer on composite panels be removed either by grinding or acid cleaning before testing or further consolidation.

High interstitial levels in titanium aluminide MMCs have been found to have negative effects on matrix ductility and toughness. The interstitial levels of incoming powders must be sufficiently low that the approximately 200 ppm oxygen added during the plasma spray process does not impair the inherent ductility and toughness of the alloy. Low interstitial content powders or more interstitial tolerant alloys must be used.

With respect to mechanical properties, HIP consolidation conditions may affect the composite yield strength by causing residual stress differences. Modeling residual stresses as a function of HIP consolidation parameters should be investigated. It would also be useful to explore other mechanical properties which may be sensitive to HIP consolidation parameters; for example, high temperature tensile strength, fatigue or creep behavior.

While microstructures have been examined with optical microscopy, it may be necessary to use transmission electron microscopy to more fully characterize the structures obtained from different HIP conditions, as well as structures formed by subsequent heat treatments.

1.0 Introduction

The goal of Concurrent Engineering (CE) is to shorten the development process by allowing engineers to address diverse life cycle issues in parallel to avoid long redesign efforts. Cimflex Teknowledge has taken the approach of providing tools to address requirements for functional design, physical design, manufacturing, test, reliability and maintainability concurrently, starting at the systems design stage. We have developed a Design for Testability (DFT) Advisor to demonstrate this capability in tracking the test perspective during hierarchical design. Specifically, our research addresses the following questions:

- Which testability problems can be identified early in the design cycle?
- How can Design for Testability suggestions best be presented to designers?
- Are estimates of the cost and benefits of Design for Testability suggestions useful for decision making?

In this report we describe the work completed at Cimflex Teknowledge as part of the Phase 2 effort on the DICE program. The DFT Advisor consists of three separate modules which provide a continuum of test advice for a hierarchical design as it evolves from initial requirements through detailed design. These modules are integrated with the DICE Concurrent Engineering framework.

1.1 *Problem Statement*

Test development for digital systems is becoming an increasingly difficult and expensive task. Test planning involves the selection of a testing strategy for the system, determining if the appropriate resources are available for carrying out the test strategy, and determining a test approach for each component of the system. Test strategy selection and test plan development are currently manual tasks which are done based on past experience. This requires the use of scarce expertise, and is becoming increasingly difficult as the complexity of systems grows. Moreover, tracking the test planning process is not addressed in current industrial settings.

Difficulties in test development significantly delay initial production and adversely affect maintainability and logistics. A test program can easily consist of thousands or tens of thousands of test vectors. Test development has the following problems:

- Developing a comprehensive test set that can detect all physical faults is a time-consuming process.
- Determining how good the test set is (test grading) can only be done after completion of detailed gate-level design.
- Fault simulation, the process of determining the outcome of applying tests under varying physical fault assumptions, requires considerable computing resources; and
- Design changes to improve fault detection are very expensive at later stages of development.

These problems arise because testability concerns typically are not addressed early in the design phase. Current testability analysis occurs late in the design process, that is, the gate

or logic level. If any testability problems are identified at this point, it is usually too expensive to do a redesign to improve testability. Typically, some compromises are made which result in the fielding of an inadequately tested system. On the other hand, no efficient method has been adopted which addresses testability concerns at an earlier stage. Almost all of the commercial software which support test related activities are oriented towards the detailed design stage and beyond.

1.2 Approach

The approach we have taken to support Design for Testability (DFT) concurrently with Design for Functionality (DFF) is to provide the different types of advice relevant to different phases of the design process. Figure 1 illustrates the approach of the Cimflex DFT Advisor to provide Test Plan Development assistance concurrently during Design.

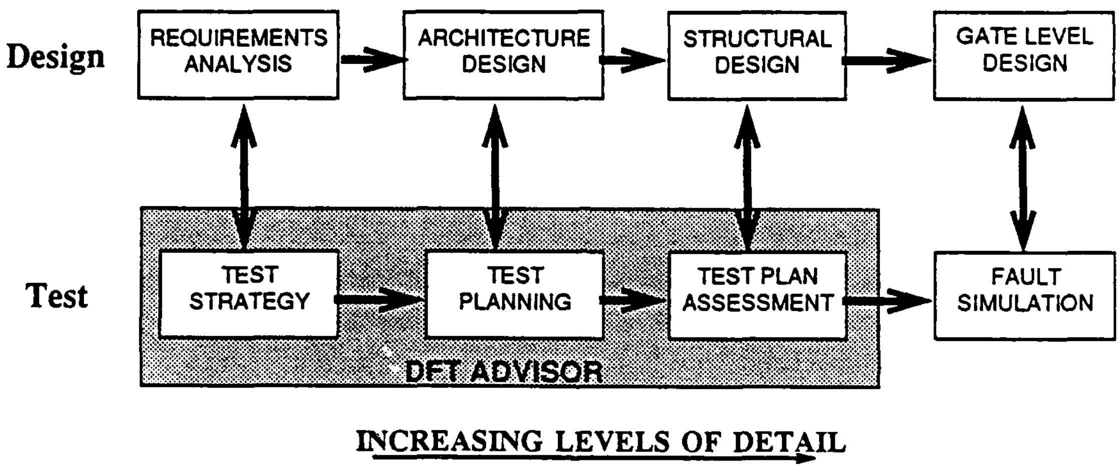


Figure 1: Concurrent Test Plan Development During Design

At the "early" design stages of Requirements Analysis, Conceptual Design, and Structural Design, the DFT advisor provides feedback on design decisions which affect the testability of the design. A desirable set of functionalites to provide this feedback include the following:

- design representations to support testability analysis concurrently with design for functionality,
- techniques which allow early identification of testability problems,
- techniques which can provide test strategy advice,
- techniques which will help the designer with concurrent test planning,
- measures which quantify and characterize testability problems early in the design phase,
- knowledge bases of techniques to improve testability of incomplete designs, and

• techniques to trade-off testability intelligently against other considerations.

Cimflex Teknowledge has developed a prototype *Design for Testability* advisor for Printed Circuit Board designs which incorporates several of these functions to give early testability feedback to designers. This report describes the current state of the DFT Advisor. Subsequent work will produce an alpha-site version based on the prototype deliverables. The lessons learned during the development and deployment of the DFT advisor will serve as the basis for developing other such advisors, eg., Design for Maintainability, Design for Reliability.

2.0 Design For Testability Advisor

The Design for Testability (DFT) advisor consists of the following three major subsystems:

1. Test Specification Generator (TSG),

2. Test Planner (TP), and

3. Test Plan Assessor (TPA).

The following subsections describe the functionality and the implementation status of each of these modules. The prototype modules have been tried out on a 1750 microprocessor based Embedded Standard Avionics Processor board (ESAP) developed by the US Navy, Naval Avionics Center, Indianopolis.

2.1 Test Specification Generator

The increasing densities of current electronic systems are continually reducing the accessibility of electronic component assemblies for testing. Packaging requirements are limiting the accessibility of the parts of the circuit for test application and the choice of test application strategy. Suitability and availability of Automatic Test Equipment (ATE) and supporting software will be significant factors in the choice of a test strategy. Test development costs, test application time, and test grading are also dependent on such resources. Development of new testers or acquisition of ATE must be planned well in advance of production. By developing a Test Specification consistent with requirements and available resources, the system designers will reduce the risk of developing a design which is hard or expensive to test.

The goal of the Test Specification Generation module is to help designers select a test strategy consistent with the customer requirements and project constraints. Designers will need to decide which test techniques (e.g., edge testing, in-circuit testing, built in testing, or combinations of these) will meet test goals and other project requirements. This choice will depend largely on packaging decisions determined from the requirements and the availability of Automatic Test Equipment (ATE) and supporting software. The knowledge about local resource availability is contained in a Test Knowledge Base maintained by test engineering personnel. The test strategy will be chosen to minimize the cost of meeting test requirements and obtain maximum test coverage using available resources.

Figure 2 shows the TSG module interfaces. System requirements are maintained in the ROSE database [Hardwick, *et. al.*, 1989] by the Requirements Manager [Fiksel and Hayes-Roth, 199C]. The TSG module communicates with the ROSE database to obtain requirements which impact testability through the LCM module developed by West Virginia University [Cleetus and Uejio, 1989].. These interfaces are discussed in detail in section 3. The

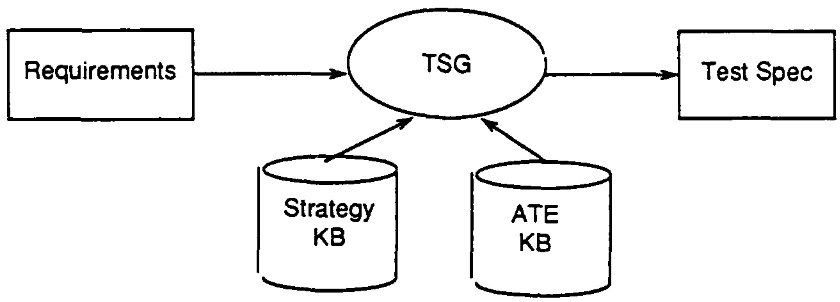


Figure 2: Test Specification Generation Module

knowledge about test equipment capabilities and local resource availability is contained in the ATE Knowledge Base. The Strategy Knowledge Base contains the test techniques and generic strategies which can be employed in board level testing. The TSG module stores the test specification back into the ROSE database. Currently the user selects the test strategy consistent with requirements. This module is implemented in Objective-C. Figure 3 shows a sample of data used by the TSG module.

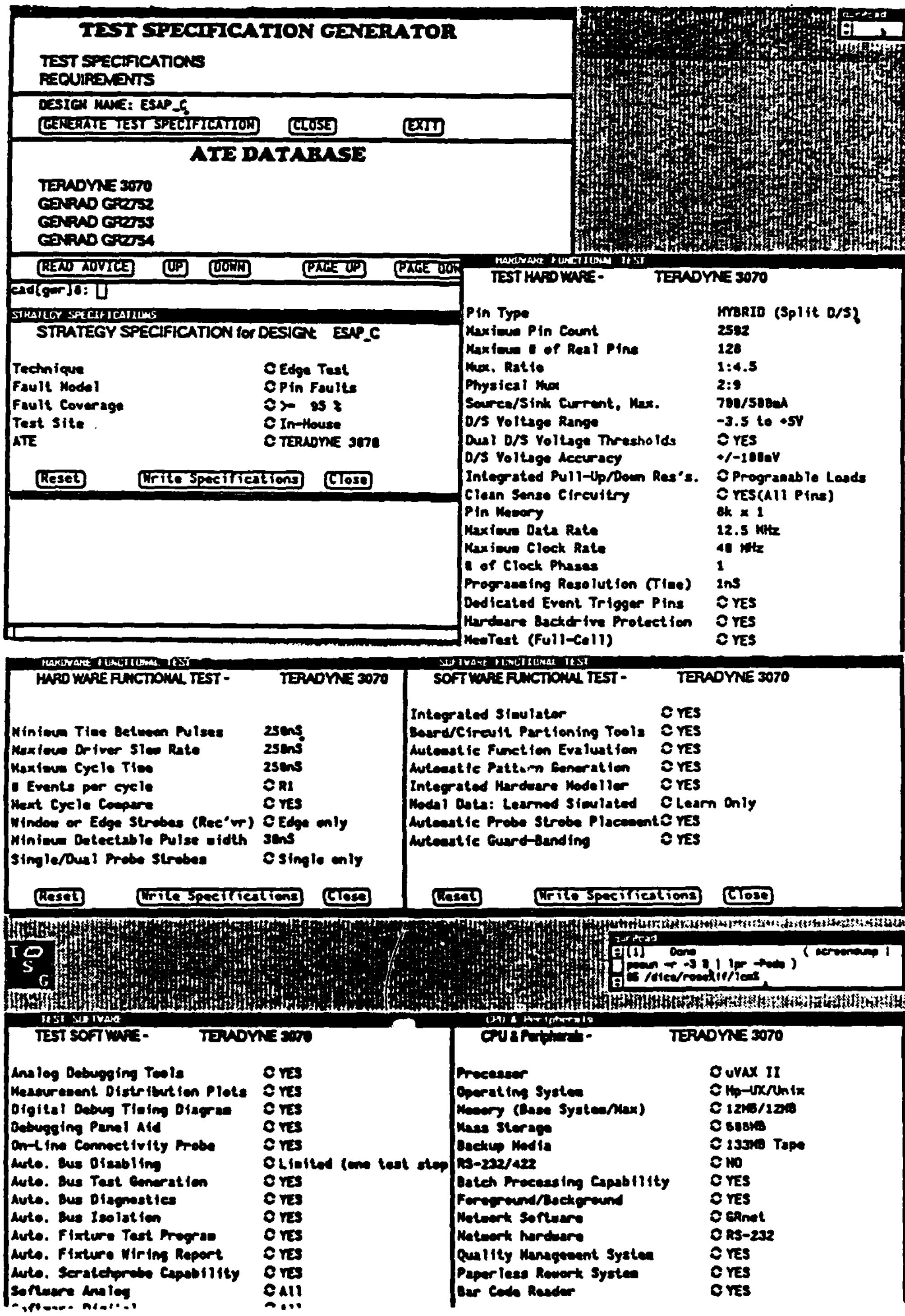


Figure 3: TSG Module

2.2 Test Planner

A flexible approach to Design for Testability (DFT) requires finding alternative ways to test the components of a hierarchical design, especially during the early stages of the design process. The designer can then make intelligent trade-off decisions among feasible testing approaches without being committed to a specific DFT method or spending large amounts of computational resources analyzing the various alternatives. The *Test Planner (TP)* module will suggest *test templates* for blocks of a hierarchical design based on a given *test specification*. The set of templates selected for the design components are analyzed for dependencies and the ordered set forms the *Test Plan*.

The Test Planner takes a test specification (produced by the TSG module) as input and uses a knowledge base of DFT Methods and associated test templates (see Fig. 4). The

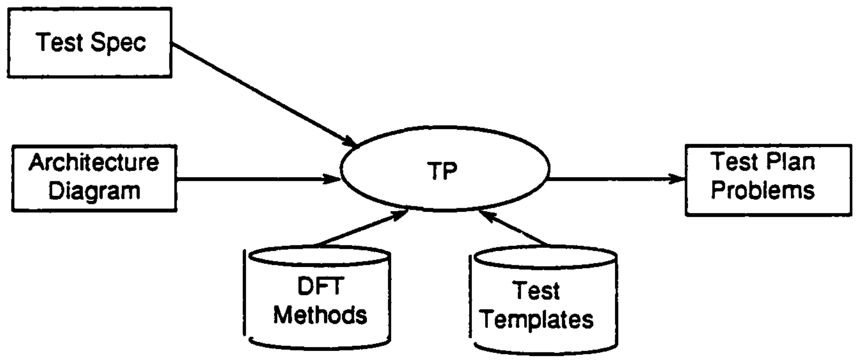


Figure 4: Test Planner Module

test specification contains testing requirements like the selected ATE types, test techniques, and test generation method, and design constraints like acceptable area overhead and signal delay. Figure 5a shows a sample specification for the ESAP board. Attributes of each DFT Method are checked using fuzzy matching to determine if the method is consistent with the test specification. Figure 5b shows the result of pruning the DFT methods for the ESAP board. The candidate test templates are those associated with the pruned set of DFT Methods.

Test templates are similar to the notion of Testable Design Methodologies (TDMs) of [Abadir and Breuer, 1985] and [Zhu, 1986]. These consist of a Source (producing the tests), Driver (applying them), Kernel (block under test), Receiver (monitoring the output), and Evaluator (detecting deviations from the expected value), as well as the paths which connect these components. The Test Planner examines candidate templates to find those which match individual blocks in the design. Figure 6a shows the user selecting a test plan template for a block in the ESAP board. All relevant templates are displayed to the user, and the reasons for the inapplicability of other DFT methods can be shown. The user can now alter the design to make desired templates applicable. As the user selects a template for each block, it is examined for dependencies with other templates; e.g., a microprocessor selected as a driver in one template should itself be tested first. These templates and the dependencies constitute the Test Plan. Figure 6b shows a partial test plan for the ESAP board.

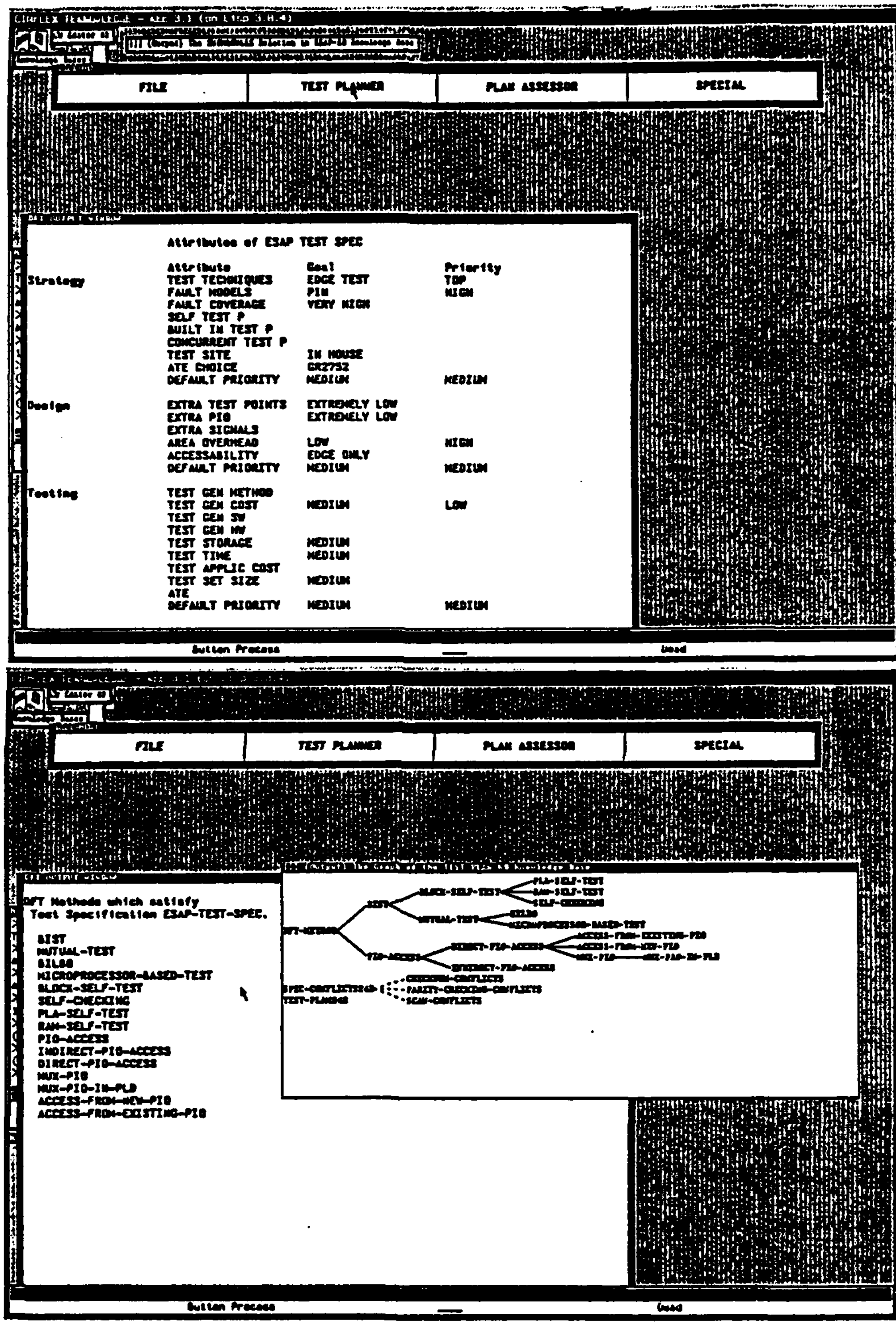


Figure 5: TP Module a) ESAP specification; b) Pruned set of DFT Methods

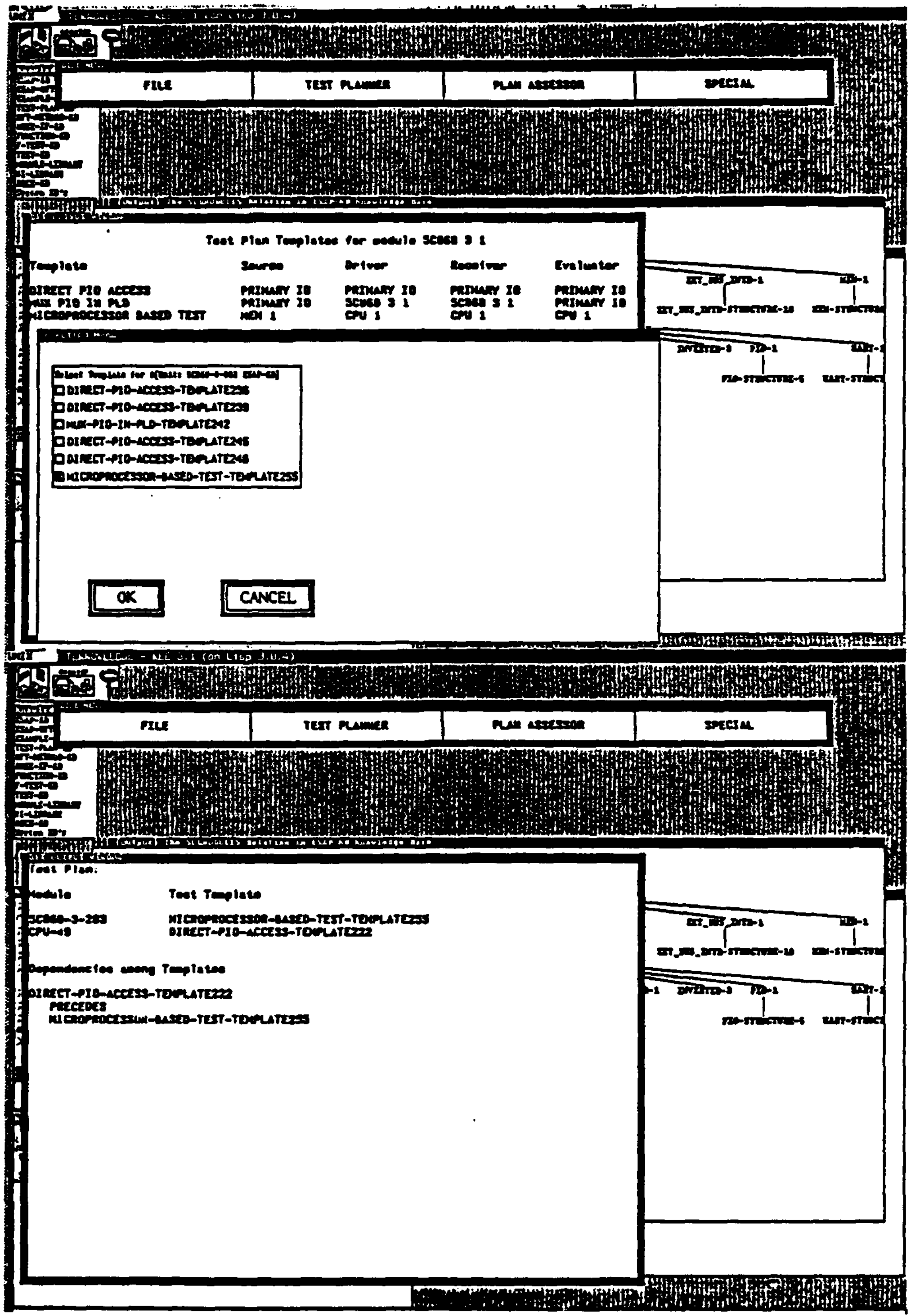


Figure 6: TP Module: a) Template selection; b) Partial Test Plan.

2.3 Test Plan Assessor

The Test Plan Assessor (TPA) module provides quantitative metrics for evaluating the test plans from the Test Planner. Figure 7 shows the interfaces for the TPA module.

The Test Plan Assessment module takes as input the current evolving state of the design, a specific module and test plan for that module. It is organized as a collection of methods, where each method estimates the value of a metric for the test plan. The current set of metrics include estimation of degree of controllability, degree of observability, test application time and fault coverage. The fault coverage, controllability and observability analysis depend on the module under test being a pre-analyzed library module and also rely on a behavioral model being available for all modules which constitute the design. The system takes advantage of detailed test data associated with modules available in a library in providing these estimates. Figure 8a shows the format used for representing the fault dictionary associated with a given module. Figure 8b shows the estimated fault coverage under assumptions of bringing out to the edge a variety of output pins for a specific block.

The Test Planner and the Test Plan Assessor are implemented in Common Lisp and KEE, a knowledge engineering environment from IntelliCorp. These modules interface to Valid's fault and logic simulation tools, and to the ROSE database through the LCM interface. The interfaces to the DICE architecutural elements are discussed in greater detail in the next section.

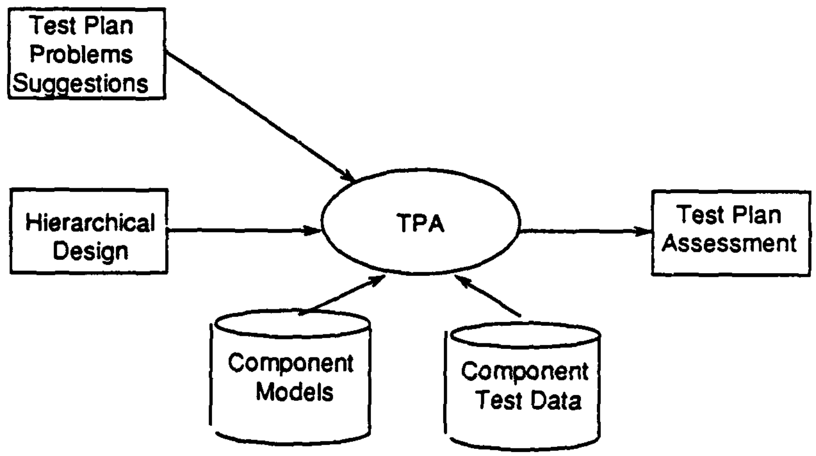


Figure 7: Test Plan Assessment Module

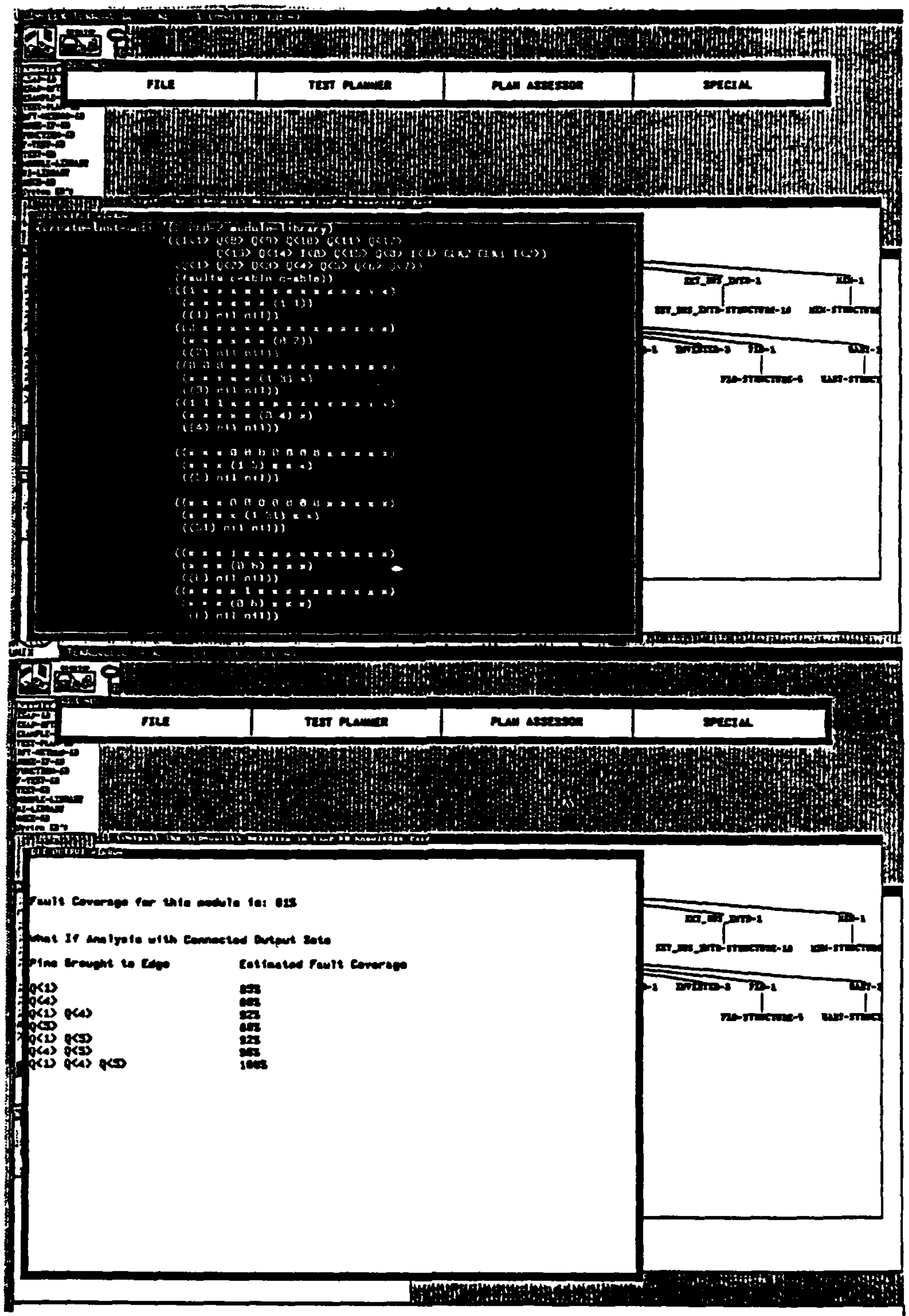


Figure 8. TPA Module: a) Fault Dictionary; b) What-if analysis for a module.

3.0 Integration with the DICE framework

The DFT prototype modules developed were tied in to the DICE framework through the

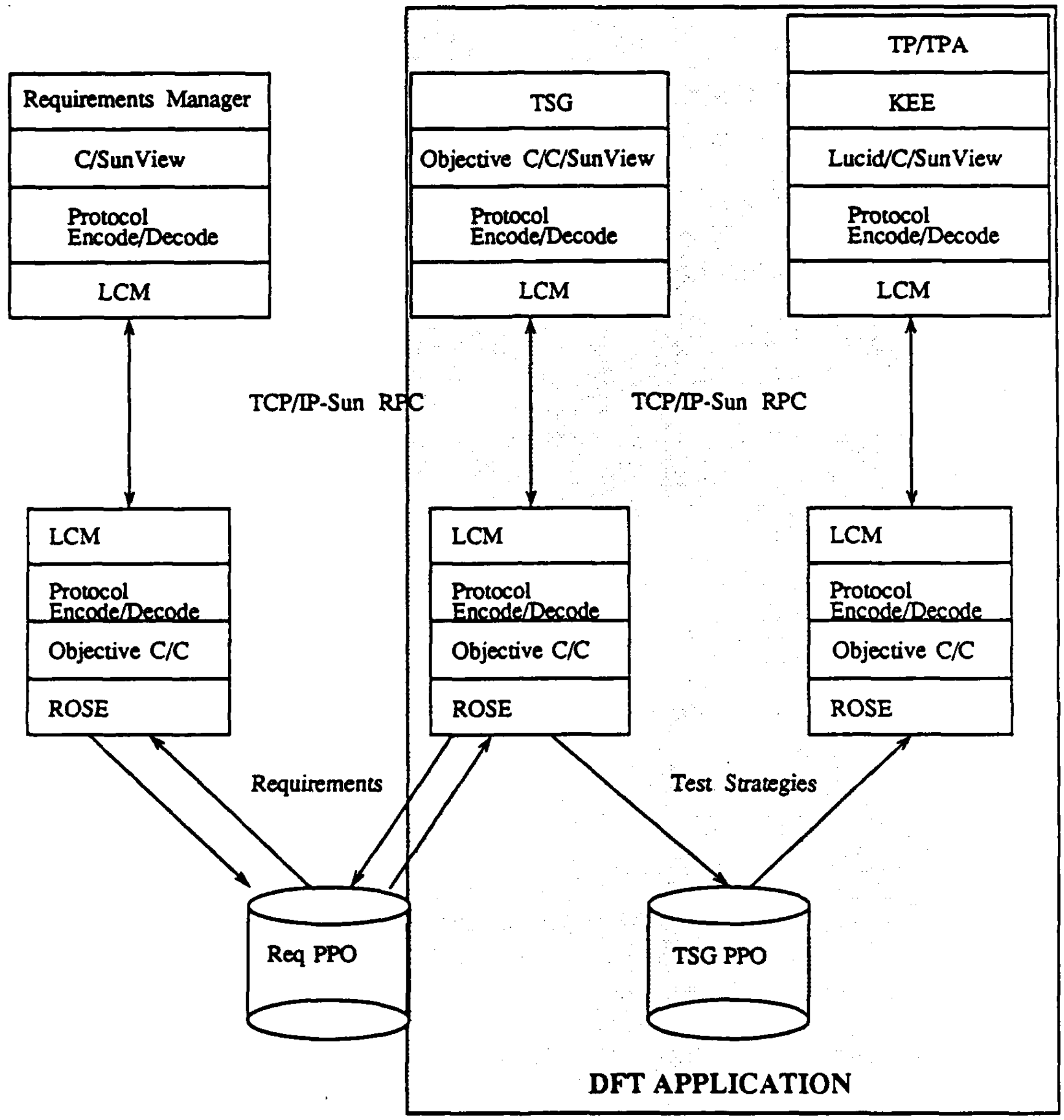


Figure 9: Integration of DFT application with DICE architecture

ROSE database (Hardwick *et. al.*, 1989) and the communication services provided by DICE. This integration involves software developed in C, Objective C, KEE and Common Lisp. Figure 9, shows the overall system view of the modules integrated with the Design for Test advisor. The primary focus of our effort in this integration was to develop the protocol to encode and decode which allows the diverse software modules to exchange information. As

part of this effort we also had to design a representation of the information to be maintained in the ROSE database, through the PPO interface. The next sections briefly address what was done in the area of defining the PPO models and in defining the protocol for encoding and decoding information passed between the various software modules.

3.1 PPO Model Definition

The PPO model defines the organization of information in the ROSE object-oriented database. To facilitate the definition of the PPO model, GE CRD has developed a preprocessor called the PPO Schema Generator (Lewis & Sarachan, 1990) which took structured English descriptions of the PPO model as input and generated PPO objects as output, expressed as Objective C class definitions.

For the DFT prototype, the PPO model contained two primary objects: a Requirements object and a Test Strategy object. Instances of the Requirements object are shared by the Requirements Manager (RM) and the TSG. Instances of the Test Strategy object are shared by the TSG and TP. The objective of this scheme is to implement the concurrent accessibility of data between various components of the electronics design, layout and test phases. These particular objects were isolated and implemented in order to demonstrate the concept of control and data flow as well as concurrent access in the concurrent engineering prototype.

The use of the PPO Schema Generator preprocessor by the TSG module to store the class definition test specification highlighted several problems in the current implementation of the Schema Generator. The interface did not allow for object-to-object reference, the information maintained had to be fairly static, and it did not allow for default values for attributes or allow multiple inheritance. But it did provide for an easy and simple interface to store information in ROSE.

3.2 Protocol Definition

The *object-to-object protocol* which had to be developed defines a protocol specification in *Abstract Syntax Notation (ASN.1)* within the ISO Open Systems Interconnection (OSI) seven-layer model (ISO OSI, 1987). The layers addressed with this solution are layer six, the *presentation* layer, and layer seven, the *application* layer. The data delivery mechanism needed to implement this protocol is provided by the DICE Communication Channel (DCC) The DCC consists of: TCP-IP based Network File System (NFS) and the *DICE Local Concurrency Manager* (LCM) (Kannan, 1989). The LCM provides point-to-point message passing service.

In Figure 9, this protocol definition correspond to the blocks termed *Protocol Encode/Decode*. The protocol encode/decode feature effectively translates data from a representation understandable by the application program and the PPO module into a uniform representation which is readily transmitted between two communicating processes. The term *encode* means to translate data from the application to the protocol format. Likewise, the term *decode* means to translate the data in the protocol format to the representation indigenous to the application. For example, the changes in format for the TP application communicating with its PPO server go like this: LISP->LCM Protocol->Objective C. For the

RM module; C->LCM Protocol->Objective C. The primary advantage of this scheme is that only one external representation is required.

The strategy adopted to develop this protocol was to take the common subset of features supported by KEE, Objective C and ROSE and develop a protocol to address that subset. The functionality covered by the protocol falls into three subdomains 1) *File I/O* - load, save and flush; 2) *Object management* - create_instance, get_value, put_value; and 3) *Queries* - applications and instances. This protocol provided the basis for the TSG and TP modules to communicate with the PPO definitions stored in ROSE.

4.0 Accomplishments for Phase 2

The key technical accomplishments of our Phase 2 DFT application effort were:

1. Identification of information requirements to allow early determination of Automatic Test Equipment (ATE) needs and overall test strategy (combinations of test techniques). Test Specification Generation is designed to keep Designers appraised of overall testing constraints and allow requirements trade-offs to improve testability versus other design goals.

2. Implementation of a prototype Test Planner using knowledge base of Design for Testability (DFT) methods and Test Plans. The TP assists a designer in development of test plans early in the design process. Designers can select from multiple approaches to satisfy overall system test requirements. The Test Planner provides a mechanism for a Test Engineer to get design intent from the designer's description of the Test Plan.

3. Development of metrics to evaluate the test plans. Controllability and Observability estimates are given separately to allow the designer to select which needs improvement. Designers can do trade-off analysis on the different ways to test the design.

4. Tracking evolving standards VHDL, EDIF, PDES. Our current design representation is based on the VHDL standard.

5. Interfaces to Commercial CAE Tools. Interfaces were developed from the DFT to the Lasar Fault simulation package anu also Valid's logic and fault simulation tools.

6. Integration of DFT with other DICE Architecture tools and environments such as ROSE, LCM and PPO. We developed a prototype integrated DFT based on the definition of an object-to-object protocol. The integration effort allowed identification of significant software bugs and resulted in more robust architecture utilities and services.

5.0 Summary

The key objectives for our effort in DICE Phase 2 were:

- Develop knowledge representation for HDE Design and Test.
- Develop technology to improve the Testability of HDE design.

Our overall accomplishments on addressing these objectives are:

- The representation issue is central to the notion of building a Knowledge-Based DFT Advisor. The representation developed applies proven AI techniques for the capture and use of Test knowledge. The integrated Design and Test representation was clearly demonstrated during the December 1989 demonstration and at the CE Symposium in February 1990. The knowledge base of test techniques and test information developed in Phase 2 is being used to determine the functional specification of the DFT advisor system which will be deployed to alpha site customers at the end of Phase 3.
- We developed a range of solutions, exceeding goals set at the beginning of Phase 2. The approach supports coordination of project management, design engineers, and test engineers through various phases of a concurrent engineering development.
- We exercised with our DFT application significant sections of the DICE architecture and provided real feedback on the state of architectural services.

6.0 References

Abadir, M. S., and Breuer, M. A., "A knowledge-based system for designing testable VLSI chips," IEEE design & test of Computers, August 1985, pp 56-68.

Cleetus, J., and Uejio, W. (eds.), "Red Book of Functional Specifications for the DICE Architecture," DICE Technical report - Draft, GE, Feb 1989.

EIA 1393a Working Document. *Manufacturing Message Service for Bidirectional Transfer of Digitally Encoded Information. Part 2: Protocol Specification.* April 1986.

Fiksel, J., Hayes-Roth, F. *Coordination of Requirements Management in Printed Circuit Board Design and Production.* Second National Symposium on Concurrent Engineering. Morgantown, W. V. Feb. 1990.

Gorlen, K., Orlow, S., Plexico, P. *Data Abstraction and Object-Oriented programming in C++.* John Wiley and Sons. ISBN 0471 92346 X. *In Press.* 1990.

Hardwick, M., Spooner D. L., Downie, B., Hvannberg, E., Loffredo, D., Mehta, A. *ROSE, A Database for Cooperative Work and Concurrent Engineering.* User Manual Version 1.0, October 10, 1989.

Intellicorp, *KEE Reference Manual Version 3.0.* July 25, 1986.

ISO OSI: *Open Systems Interconnections Basic Reference Model.* International Standard 7498, Geneva, Switzerland, 1987.

ISO *Specification of Abstract Syntax Notation 1 (ASN.1).* International Standard 8824. Geneva Switzerland, 1987.

Kannan, R. *Local Concurrency Manager User Interface Manual.* West Virginia University. 1989.

Matsumoto, A., Jagannathan, V., Saks, V., Buenzli, C. *Concurrent Design for Testability.* Workshop on Concurrent Engineering Design, International Joint Conference on Artificial Intelligence (IJCAI), Detroit, MI, August, 1989.

Matsumoto, A., Jagannathan, V., Saks, V., Buenzli, C. *Concurrent Design for Testability of Hierarchical Boards.* Second National Symposium on Concurrent Engineering. Morgantown, W. V. Feb. 1990.

Pelton, G. A. *Application Architecture and Application Interface Specification for Manufacturing Floor Environments.* Proceedings of IECON, Milwaukee WI., MAP Subgroup, Oct. 1986.

Sarachan, B. D., Lewis, J. W. *PPO Schema Management.* Second National Symposium on Concurrent Engineering. Morgantown, W. V. Feb. 1990.

Snyder, A. *Encapsulation and Inheritance in Object-Oriented Programming Languages.* Proc. ACM Conference on Object-Oriented Systems, Languages and Applications. Portland, Oregon, Sep. 1986. 38-45.

Stefik, M., Bobrow, D. G. *Object-Oriented Programming: Themes and Variations.* The AI Magazine. Vol. 6, No. 4. Winter 1986. 40-62.

Stroustrop, B. *The C++ Programming Language.* Addison-Wesley, Reading, MA. July 1987.

Stepstone Corporation, *ICpack 101 User Reference Manual Release 4.0.* 1988.

Zhu, X., "A knowledge-based system for testable design methodology selection," *Technical report CRI-86-23*, University of Southern California, August 1986.

DICE Program

Task 4.4.7.2 Design For Assembly Advisor **West Virginia University**

Objectives:

The overall objective of this task was to develop a computer-based Design For Manufacturability/Assemblability (DFM/A) module for the DICE Design For X workstation. This DFM/A module will, for example, enable the lead designer to perform concept-level evaluations between the physical design specifications and the plant production operations early in the concurrent engineering product and process development cycle (see Figure 1). The primary focus of the module development is to enhance the assemblability of the electronic printed circuit board (EAR, IPP, SSP, and TPS module) as well as its manufacturability (PWBMA module). Module development was also pursued on enhancing the assemblability of mechanical systems (IMAD module) (see Figure 2 and Table 1).

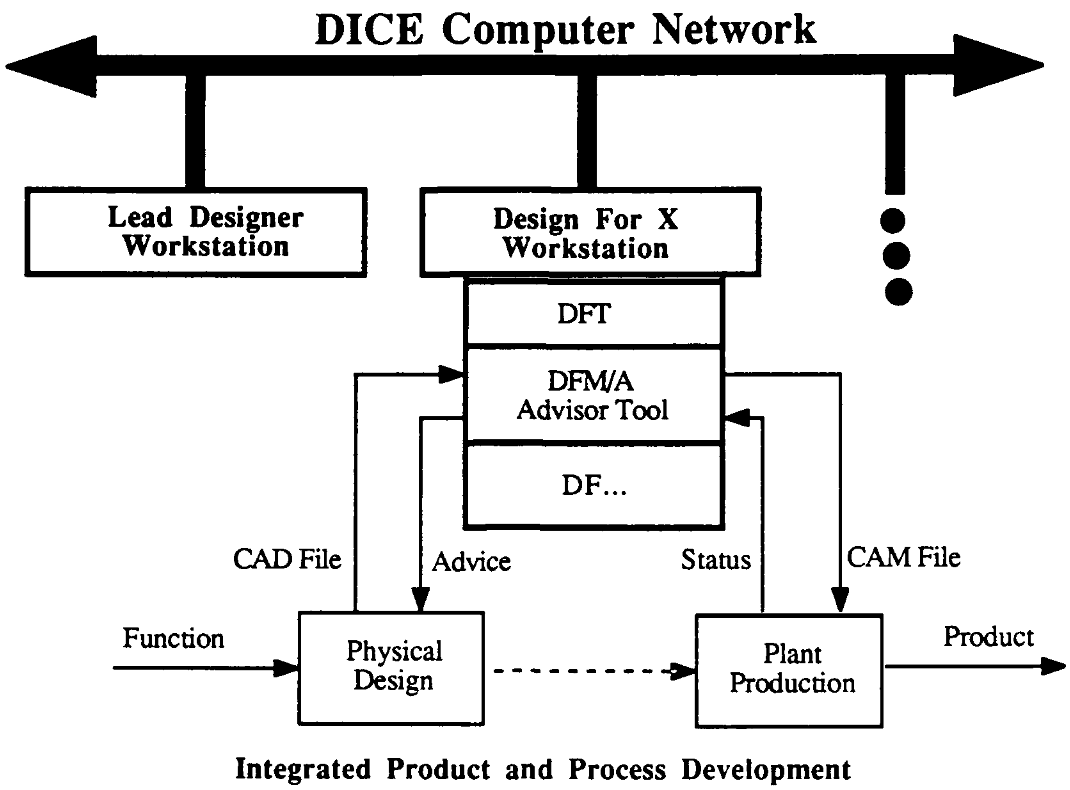


Figure 1. The DFM/A Module in the DICE System

The advisory aspects of the DFM/A module will permit the user of the module to address and/or resolve design problems such as (1) enabling the procedure for the optimized assembly of mechanical systems to be graphically accomplished; (2) enabling the modifications to the design parameters for printed circuit board manufacturability to be expertly suggested; (3) enabling the evaluation of printed circuit board design and defects based on rating information and cost estimation to be dependably provided; (4) enabling the translation of a CAD-based layout file for a printed circuit board into an efficient workcell controller file to be quickly achieved; (5) enabling the placement sequence of components on a printed circuit board for a specified assembly workcell configuration to be spatially optimized; (6) enabling the robot motions for printed circuit board assembly considering collision detection, minimum paths and trajectory speeds to be graphically simulated; and (7) enabling the programming time for

achieving assembly workcell production of printed circuit boards to be drastically reduced.

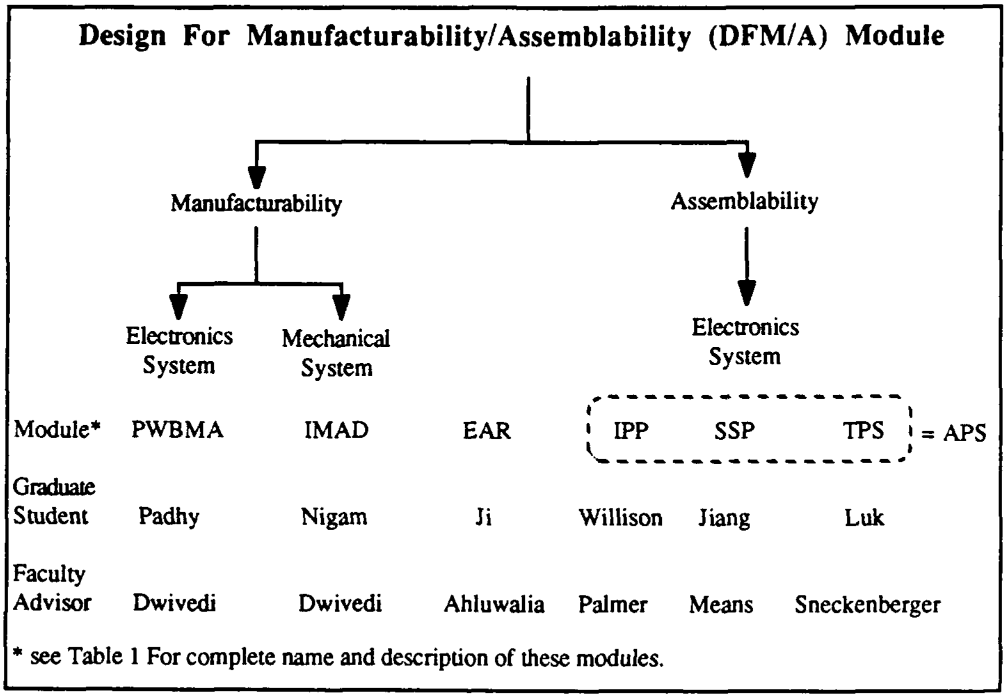


Figure 2. The Modules inthe DFM/A Module

Approach:

The general method used to develop a DFM/A module was to develop specific modules for particular manufacturability and assemblability purposes, as shown in Figure 2. Primary emphasis was placed on modules related to electronics systems and, more precisely, the printed circuit board (PCB).

Table 1
Module Names and Descriptions:

- PWBMA.......Printed Wire Board Manufacturability Advisor

  An advisor for the manufacturability of printed circuit boards that uses a rule-based expert system to provide interactive advice to the board designer based on specific board geometric and functional requirements.

- IMAD...........Intelligent Mechanical Assembly Designer

  An advisor for the assembly of mechanical systems that uses a rule-based expert system to provide advice interactive to the designer based on developed guidelines for efficient mechanical designing using solid modeling features.

- EAR..............Electronic Assembly Ranker

  An advisor for the assembly of printed circuit boards that uses a rule-based expert system to provide interactive advice graphically to the board designer based on assemblability guidelines and cost estimations.

- IPP................IGES Pre/Postprocessor

  An advisor/tool for the assembly of printed circuit boards that uses a file transfer to generate a board component location and orientation file and a board assembly workcell file from a CAD-based board input file.

- SSP................Special Sequence Planner

  An advisor/tool for the assembly of printed circuit boards that uses an optimization method to provide the best feasible sequence for assembly of the board components in terms of spatial workcell considerations.

- TPS...............Trajectory Planner and Simulator

  An advisor/tool for the assembly or printed circuit boards that uses collision detection and trajectory planning algorithms to provide simulated manipulator path and motion mechanics for board assembly by a robot assembly workcell.

- ASP...............Assembly Planner and Simulator

  An integrated module for assembly of printed circuit boards that consists of the IPP, SSP and TPS modules with networked exchanges of files and messages to provide the most productive workcell assembly of a given CAD-based board.

The PWBMA module is being developed as an expert system with rules and guidelines for the manufacturability of the printed circuit board. The parameters affecting etching, screen printing, masking, board layers, drilling of material and its effect on the material, board layout as well as other important parameters are critically analyzed and incorporated into the knowledge base. The expert system is built upon the VP-Expert shell. The system is user friendly and able to suggest modifications to the design parameters for improved manufacturability.

The IMAD module is being developed using principles similar to the Boothroyd-Dewhurst principles. In addition, an optimum configuration algorithm will be generated for assemblability. Modification advice for the better design of components for assembly will be also incorporated in the system. The suggested assembly design modifications will be displayed using a visual representation.

The EAR module uses a board rating package based on assembly rules and guidelines. Each rule carries a certain weight, or relative importance, and components earn a certain rating value depending on the features of the board being assessed. The module considers the following assembly areas: automatic assembly compatibility, manual operations requirements, testability, solderability, cleanability, inspectability and performance. The program also

provides a cost estimation. Given the rating information and cost estimation, the designer can evaluate the board design and trace design flaws.

The IPP module uses data extraction techniques to create internal files to provide geometric layout definition for a PCB design as well as component name and dimensions.

The SSP module uses the linear implicit enumeration method. It assumes the board and part locations are fixed by the designer. The approach is to generate the optimized part assembly sequence. A non-linear optimization program can evaluate the optimum feeder locations by optimizing the total robot displacements. The results are then used as the input data for the linear implicit enumeration program and the optimized assembly sequence is iteratively found.

The TPS module, as advisory enhanced during this phase, evaluates PCB layouts by detecting inappropriate assembly operations caused by undesired assembly circumstances, such as impossible assembly sequences or overlapped component locations. The module can also evaluate user-specified assembly requirements, (e.g. maximum board assembly time,) and provide advisory messages if any violation of these specified requirements is detected during the simulated assembly.

The APS module was the initial effort to integrate the individual manufacturability and assemblability modules into a DFM/A module. This module integration was achieved through the development of appropriate computer hardware and software networking capabilities. These developments included network provisions to enable personal computers and workstations to share data and messages interactively.

Technical Results:

The technical accomplishments during this phase of the DICE program focused mainly on the incorporation of expert system capabilities in several modules and the integration of assembly planning and simulation modules into an integration module.

The PWBMA module is an expert system that works in an interactive way. The designer provides design data or parameters to expert system queries in order to perform the manufacturability evaluation. The technical result can be

described as a design aid for designers in manufacturing the board in a most efficient way. Design modification and suggestions to the designer that will enhance the productivity by elimination of more rejections, by reduction of time and by reduction to overall cost of manufacture, are generated through the module. The module can work as a standalone system with a single user interface or can be integrated in a network system with multiple data interchanges possibility. The expert system also provides a new designer with the capability to learn the important considerations of board manufacturability.

The IMAD module also works as an expert system. This expert system was developed using the VP-expert shell and assembly rules and guidelines. The module is capable of advising the designer whether or not the component is designed for assemblability. A mechanical assembly is ranked using individual component rankings. Graphic assembly designs using software packages like IDEAS, ICAD and PostScript can also be investigated.

The EAR module developed a method to rank order the assembly of a printed circuit board based on an assembly advisor. The research was concerned with the algorithms and methodologies for application of AI to PCB assemblability and the integration of such hardware and software. The object-oriented expert system LASER was used to handle all rules and guidelines of assembly. The expert system interfaced with the computer-aided design software so that the PCB design could be modified and re-evaluated.

The IPP module takes a file from the printed circuit board CAD package and extracts the data needed for board assembly by the workcell. The data extracted is sent to the SSP and TPS modules for assembly optimization and trajectory planning. In addition, an output file is generated for automated workcell assembly of the PCB. Coordination with workcell manufacturer was done to allow inspection of the workcell using a sample workcell file.

The main function of the SSP module is to provide an evaluation of the components on a printed circuit board in an automated workcell in terms of the optimum assembly sequence based on minimum spatial distance traversed by the assembly manipulator. An implicit enumeration method is used to accomplish this evaluation. The optimization software, which was written to run on a mainframes computer system or on a personal computer, is now installed on a workstation.

The SSP module has four input data files. It shares the workcell input data with the TPS module. The PCB data file is from the IPP module. The workcell file

contains the coordinates of the origin of the PCB with respect to the location of camera. The fourth input data file contains the constraint equations on optimization. The file is automatically written by a program, which only needs the numbers of parts and feeders. The output file provides the optimized sequence in the format which the TPS module can use.The optimization capability of the SSP module increased from 15 to 20 board components while the number of feeders increased from 3 to 4.

The TPS module provides a graphical simulation that animates the PCB assembly operations of a robotic assembly workcell. The module consisted of a collision detection algorithm, a minimum path searching algorithm, and a robot trajectory planner.

During this phase, the internal data structure of the TPS module was modified to be efficiently integrated with the IPP module and the SSP module as a part of the APS module. The robot trajectory planning capability was upgraded by introducing an additional trajectory planner called the parabolic blend trajectory planner. A faster robot trajectory can now be planned that is particularly good for straight line robot motion planning. Assembly workcell error checking and user-specified requirement confirmation was also implemented. The collision-free path searching algorithm was enhanced to handle a larger PCB layout by using linked list data structure in the A* search algorithm. The number of board components that the TPS module can manage increased from 20 to at least 60. Some graphical user interface, including push buttons, dialog boxes and scrolled bars, was implemented to provide the user more control to the module and a more user friendly working environment.

As a participant in the DICE Electronics Domain Demonstration in December, 1989 , the APS module concept provided the PCB scenario with an advisory tool for evaluating proposed board layouts as well as generating controller instruction for downloading to the board assembly workcell. The conceptual APS module was functionally completed for a possible February, 1990 DICE Demonstration.

Conclusions:

The important capabilities currently provided by the evolving DFM/A module through its six existing individual modules and its one integrated module are stated in the following important findings and conclusions for each module.

The PWBMA module is developed using a PC based expert system shell. The rules and guidelines pertaining to PCB manufacturability are incorporated in the knowledge base. The expert system is capable of providing suggestions and design modifications to the user.

The IMAD module is implemented on a PC based shell and is capable of performing mechanical assembly evaluations. The module is capable of handling various geometric parts and suggesting design modifications.

At present, the EAR module can input a PCB design and display it on the screen, and it can classify components and determine their relationships. Present component classification identifies odd-shaped components, Surface Mounted Devices (SMD) and through-hole devices. These components are matched with workcell capability. The component relationship module finds the component neighbors. This information is used for space checking. The software package can read a pseudo PCB design file and show the design in a window. The software also provides component classification. The relationship of components is found and stored in a linked list file.

The IPP module successfully translated CAD data file from IGES format to the file format that the SSP module and the TPS module in APS can use. IPP also extracts workcell information from the workcell controller. IPP generated a simple ADX workcell command file which was downloaded to the Cimflex workcell.

The SSP module uses free format and open commands to obtain different input data provided by the customer. Customers can change input data such as feeder locations, part positions and PCB locations any time, as needed. The SSP is a flexible software package. It can work for any different composition of the feeder number and the part number up to 20x20. The constraint equations for optimization are automatically written by SSP. When the input data is changed by the customer, the constraint equation will be changed automatically.

The TPS module was enhanced to become more powerful and more user friendly. The accomplishments included implementation of a 2-1-2 trajectory

planner, graphical user interface, error checking and constraint confirmation, modification of the module data structure, and integration of into APS advisory tool.

The APS advisory tool was developed as the successful integration of the above IPP, SSP and TPS modules.

Recommendations:

The capabilities of the present PWBMA module are limited due to the VP-expert shell. The problem of interacting with the system in this shell hinders the flexibility to develop sophisticated applications. Future work, including the use of object-oriented environments using Smalltalk and DICEtalk environments, is recommended. A grading system needs to be enhanced to work with design files and to link with the VALID logic software.

The IMAD module needs to incorporate a graphical output using the object-oriented shell DICEtalk. The graphical interface will be achieved using Smalltalk object-oriented language.

Future work on the EAR module should include the use of the LASER expert system to handle all assembly rules and guidelines and the input of information of an actual PCB and an assembly workcell. Additional software needs to be developed for providing a bi-directional interface with the designer.

The IPP module needs to automate the generation of the workcell command file internal to the APS module based on assembly sequence and trajectory data provided by the SSP and TPS modules.

It is recommended for the SSP module that the present linear optimization software be combined with the non-linear subroutine program to evaluate the optimum coordinates of the feeders and feedback to the designer. This will allow for the total optimization of the automated workcell. It is also proposed to incorporate the software package I-GRIP to show the assembly process and sequence in real time in three-dimensional graphics.

Specific recommendations for the TPS module are to enhance the capability of the module to be able to plan paths for more then one robot in the same assembly workcell and to enhance the module to be able to handle mechanical assembly planning.

There is an appreciated interest to extend the APS module to include a CAD-based PCB layout package and subsequently to incorporate PCB assemblability and manufacturability advisor modules.

Publications:

Jiang, J. and K. H. Means. "Optimum Assembly Methods", Fifth International Conference on CAD/CAM, Robotics and Factories of the Future. December 1990. (under consideration).

Luk, T. L. and J. E. Sneckenberger. "Trajectory Planning For Obstacle-Avoided Assembly of Planar Printed Circuit Boards." Fifth International Conference on CAD/CAM, Robotics and Factories of the Future, December 1990. (under consideration).

Luk, T. L.. "Automated Synthesis of Robot Arm Motion For Robot Arm Collision Avoidance." Ph.D Dissertation, WVU (in preparation).

Padhy, S. K. and S. N. Dwivedi. "A Rule Based Expert System for Design and Assembly of Printed Circuit Boards." Fourth International Conference on CAD/CAM, Robotics and Factories of the Future, December 1989.

Padhy, S. K., R. Sharan and S. N. Dwivedi. "Design with Feature at Conceptual Stage." Second National Symposium on Concurrent Engineering, February 1990. (under consideration)

Padhy, S. R. and S. N. Dwivedi. "On the Contact and Geometry of Features in Assembly." Fifth International Conference on CAD/CAM, Robotics and Factories of the Future, December 1990. (under consideration).

Willison, R. H.. "Design of an IGES Postprocessor and Implementation with a Robotic Workcell." MSME Thesis, WVU, (in preparation).

Hardware:

No hardware or software was purchased especially for this task during Phase 2. However, this task benefited by the purchase and installation of NFS and PC-

NFS software packages. The task also appreciated the networking and messaging software capabilities developed by the CERC Architecture Group, especially Joe Cleetus, Naresh Mareshwari and Brad Bennett which contributed to the development of the integrated APS module.

The first six software modules listed in Table 1, which were begun in Phase 1, were further developed during Phase 2. The APS software module was initiated and developed by this task during Phase 2. This module is the collected integration of the IPP, SSP and TPS modules as indicated by their dashed encirclement in Figure 2.

DICE Program

Task 4.4.8.2 Cost Modeler **West Virginia University**

Objectives:

Objectives of this task is to ascertain the suitability of existing cost estimation procedures, to develop analytical techniques for cost estimation, to develop demonstrations of a Cost Advisor concurrent engineering (CE) module and to solve a specific cost estimation problem. The background study included analysis of existing cost estimation methods and existing cost accounting procedures. Analytic techniques centered on cost improvement effects on eventual product costs. Investment casting of turbine blades was the central Cost Advisor demonstrated. Costs of laying up composite materials were analyzed and modeled into a specific Cost Advisor.

Approach:

Study of basic costing methods included a review of commonly accepted accounting methods, cost accounting procedures and cost modeling techniques. Some specific costing tools were also reviewed. The questions involved were: how are costing activities conducted?; and are extant methods of use to CE systems? The conduct of this research included the perusal of textbooks on such topics as Fundamental Accounting, Financial Analysis and Managerial Accounting. In addition, the current status of the debate over product costing methods was reviewed. An additional topic under study was the trend toward implementation of federal Design for Cost procedures. The integration of existing procedures with improvement curves and life cycle costing techniques was also studied and will become increasingly important during Phase III.

Improvement curve theory (also called experience curve theory) was also under study. This is an important aspect of new product development since it is crucial to GO-NO GO decisions to know the eventual average cost per unit of projected product output. The natural tendency of costs to decline over multiple improvement rates is a topic under literature review and mathematical analysis.

The eventual DICE application is the calculation of this kind of improvement effect into CE Cost Advisors.

The investment casting cost equations were put into a LOTUS spreadsheet to simulate interactive cost advising within a CE format. The question of what exact data inputs and outputs can be exchanged with other DICE tasks is currently under discussion. The natural way to perform these exchanges at the current stage of task integration seems to be via ASCII files using the system architecture as the file-access mechanism. The problem to be overcome is the exact form of the data exchanges. The use of the architecture services appears to be routine . The approach to the problem consisted of conversation and trial-and-error attempts.

The most applied activity performed during Phase II was the creating a Cost Advisor for a specific design problem. The approach employed in working on this problem closely parallels the approach needed at a real-life CE facility. The data under study described the laying up of composite materials on the TF-34 fan frame. Data were obtained from GE. The steps were as follows :

1. Data were analyzed using statistical techniques, such as stepwise analysis;
2. Candidate parametric cost equations were prepared from the analyses and tested against data that could be found in the technical literature. A "best" equation for cost prediction was adopted. This equation also appeared to make good intuitive sense;
3. A rough spreadsheet model for answering design queries, with costs of those designs, was prepared;
4. GE personnel reviewed the equation, variables, parameters, and the query model's structure, making suggestions from the user's point of view and
5. The requisite changes were made and the Cost Advisor was delivered to the user.

Technical Results:

The Cost Advisor described above is generically called a parametric model. This type of model requires a well-designed data gathering experiment to yield

truly useful cost predictions. The Cost Modeler staff found that there are four basic kinds of cost models employed by industrial and government users.

These four types of models and their likely utility in the CE environment follow:

1. Parametric models are ordinarily equations for estimating the costs of producing items that have an extensive history of production. The models are commonly obtained via regression analysis and are usually of little use in concurrent engineering, for two reasons: (a) the nature of the statistical "experiment" that produced the equations is unknown and (b) CE products are not likely to be replicates of existing products.
   The very nature of common, large-scale parametric estimators, such as GE PRICE, makes it unlikely they can be of use in predicting costs of new designs intended for small lot sizes. The situation described above in deriving a parametric model for the composites was somewhat better in that questions could be asked of knowledgeable personnel. This technique would be very useful if Cost Modeler personnel could oversee data collection itself;

2. Incremental models are often derived by a firm that has considerable experience producing an item it intends to revise slightly. This modeling approach is of limited value to CE cost predicting. However, the problem with employing this technique is that it requires stable estimates of prior item costs, but a CE facility is likely to make only small lot sizes of any of its products. Again, this technique will not usually be apt to suit CE needs;

3. Functional models seem to hold the most promise. Functional models usually break down a production process into constituent steps and the estimated costs at each step. Further, the available models of this type tend to refer to common processes, such as metal casting. This is the type of model employed to give cost estimates of turbine blades in Cost Advisor demonstrations. These models can be useful in a CE environment but they still require a sufficient database to adapt them to the specific CE production environment.

A common type of functional model assumes that the product to be manufactured will consist of common, readily available components that can be assembled in a particular fashion. This type of model is especially ill-suited to the level of innovation expected of a CE facility and

4. Accounting models, created as an offshoot of standard corporate accounting procedures, are odd items. Accountants accumulate a great deal of data, both in dollars and in physical quantities, that describe production processes. This is a primary source for cost estimates within most manufacturing operations. These systems have not been found suitable for accurate product cost estimates in large firms and are doubly inaccurate in dealing with the small-lot-size environment.

The Cost Modeler team experience in looking for cost models and in devising a specific cost advisor from scratch suggests that the approach employed in dealing with the TF-34 data is an appropriate one to employ at a CE site. However, there are additional considerations.

Conclusions:

Costs of products in a CE facility must be obtained differently than costs for products at a large, high-volume manufacturing firm. The special needs of the CE cost models dictate a different approach to the accounting treatment of costs and how costs are tracked. Consider R&D costs: In a large American firm R&D costs run no more than 3% of sales. In the usual large corporations' accounting procedure, R&D is lumped with diverse other costs under the category of Overhead. A substantial portion of overhead is then assigned to production as Manufacturing Overhead, but the R&D cost is not associated with the specific developing products that incur the costs.

CE cost models cannot allow overhead costs to be homogenized in this fashion. Three points establish this fact:

1. A CE facility will work as a small subunit within a larger firm or as a small independent operation and will not have large overhead;
2. R&D is too large a portion of a CE product's costs to not be directly associated with the product's cost base and

3. Lot sizes in a CE plant are likely to be so small that accurate product profitability prediction will demand a full accounting of all costs making up the product's cost base.

The implicit cost system required to track costs in a CE facility will necessarily diverge from the usual. R&D personnel will have to assign the bulk of their time and expenditure to specific designs-in-process. Engineering change order (ECO) costs will have to be assigned to specific products. Overhead, as an accounting category, must be held to a minimum so that as many costs as possible can be assigned to products as direct or indirect costs.

Changes in production, QC, and inspection costs must also be tracked to allow adaptation of cost models as costs per unit change. In particular, marked reductions in cost per unit can be expected over fairly short time periods for small-lot products. This is the costing aspect described by improvement curves.

Coupled with this research into cost models and its work on preparing a new, specific Cost Advisor, it appears that a unique framework for cost analysis is required by a CE facility. It must accumulate all costs associated with a product design so total costs can always be measured against cost criteria to make GO-NO GO decisions. Cost models must be developed from scratch or recalibrated from outside to fit the specific facility's configuration and methods.

The Cost Modeler staff concluded that it needs to draw up both: (1) an organon, as general as possible, of CE cost analysis and accounting procedures and (2) a computational template that a CE facility can actually use in preparing its cost analyses and estimates. This system incorporates improvement curve effects in the cost prediction procedures.

Recommendations:

The entire DICE Project still needs to be integrated over the complete range of its ostensible objectives--a specific illustration that will actually demonstrate data development, task interactions (concurrency), design, prototyping, testing and manufacturing prepared as a high-priority activity. The scope of this demonstration would be restricted only to the extent that it is actually achievable. This means that all relevant DICE module (task) participants will have to be assembled for coordinating discussions and will have to develop interaction (concurrent) communications.

As a result of a background analysis on existing manufacturing systems, the team believes that the DICE project should develop an instructional manual for startup concurrent engineering facilities on how to approach cost modelling and analysis. However, this manual cannot be prepared without extensive experience within an operational concurrent engineering demonstration environment. Therefore, the team recommends it restrict itself to developing a generalized CE costing system that can be adaptable to most startup CE facilities.

Publications:

The Cost Modeler team published papers without the DICE aegis. These papers essentially described the economic and cost accounting foundations of cost estimation in manufacturing generally and in their role in concurrent engineering particularly. Included were two papers published in Cost Engineering and papers published in India, Paris, and San Francisco.

Hardware:

No new hardware was purchased; however, a PS/2 is now allocated to be shared with others by this task. No original software was developed; but, templates for procedures to be run on Lotus were completed.

DICE PROGRAM

Task 4.4.9.2 Optimization of Composites GE-CRD

OBJECTIVES

In Phase I of the DICE program a design methodology and a software system, based on GE R&D Center's structural optimization software DESIGN_OPT, was developed for ply optimization of 2D fiber reinforced laminated composites. Micromechanical models developed in a related DICE I task were integrated to address the mechanics issues of composites behavior. The software was successfully applied to illustrative composite structural optimization problems. The present DICE II Task was primarily aimed at enhancing this methodology to include structural shape and micromechanical parameter optimization. Specifically, a geometry-based approach was developed and implemented in the DESIGN_OPT software for both shape and ply optimization of 2D (plane stress, plane strain and axisymmetric) composites. As a part of the software demonstration, example results were developed for plane stress plates and axisymmetric rotating discs. The objective of the micromechanical parameter optimization effort was to develop a computer code MICROMECH_OPT by integrating micromechanical behavior of uni-directional plies with a numerical optimization software. Several application studies were carried out to optimize the ply properties using micromechanical parameters like the fiber volume fraction and matrix properties as optimization design variables. In addition, the micromechanical models for residual stresses due to laminating process for metal matrix composites, developed in a complementary DICE II - Task 3.4.5.2, was also interfaced with the present work and some interesting results were obtained.

APPROACH

The overall technical approach is based upon integrating numerical optimization methods, finite element analysis, CAE tools/software engineering and mechanics of composite materials. Mathematically, a numerical optimization problem can be stated as follows:

Find: $\underline{X} = [X_1, X_2, \cdots, X_n]$

To minimize: $F(\underline{X})$

Subject to: $\underline{g}(\underline{X}) \leq 0$

$\underline{h}(\underline{X}) = 0$

$X^l \leq \underline{X} \leq X^u$

Where $\underline{X}$ represents the design parameters, F the objective function, $\underline{g}$ and $\underline{h}$ the inequality and equality constraints, and $\underline{X}^l$ and $\underline{X}^u$ the lower and upper bounds on design variables. The mathematical problem stated above can be solved by using a number of available numerical methods [1] and software packages. In the present work, a state-of-the-art computer code COPES/ADS [2,3] was employed for the purposes of numerical optimization.

When using numerical optimization as a framework for composites design, the structural shape, ply thicknesses and ply angles are treated as design parameters, the objective function typically involves minimizing the structural weight, maximizing stiffness or strength, or maximizing frequency or stability margin. Design constraints are usually imposed on stresses, dynamic response including natural frequencies, buckling behavior, strength and stiffness, and structural failure including fiber delamination, fracture and fatigue. Thus, an example optimization formulation of a composites design problem can be expressed as follows:

Design Variables: 2D structural shape, ply thicknesses and ply angles

Objective Function: Minimize weight

Constraints: Stresses, frequencies, stiffness, strength

A geometry-based approach [4] was employed for shape and ply optimization as shown in Figure 1. In this, a geometric modeler consisting of lines, arcs and splines is used for shape or boundary description, an automatic mesh generator for creating the finite element model and an attribute specification code for specifying boundary conditions at the geometry level. Further, the optimization problem formulation and design constraints are specified at the geometry level rather than individual nodes and elements. Composite structural analysis is carried out by using the finite element code ANSYS and a laminate analysis code AC3 (Figure 1). In essence, given the ply thicknesses, angles and schedule, the laminate analysis Code AC3 transforms a nonhomogeneous composite laminate into a homogeneous continuum with orthotropic material constants. The orthotropic material model in ANSYS is then used to carry out the stress, frequency and/or stability analysis.

Micromechanical ply optimization consists of determining fiber and matrix properties and the fiber volume fraction to optimize a certain ply property with constraints on other ply properties. As an example, the micromechanical ply optimization problem can be stated as follows:

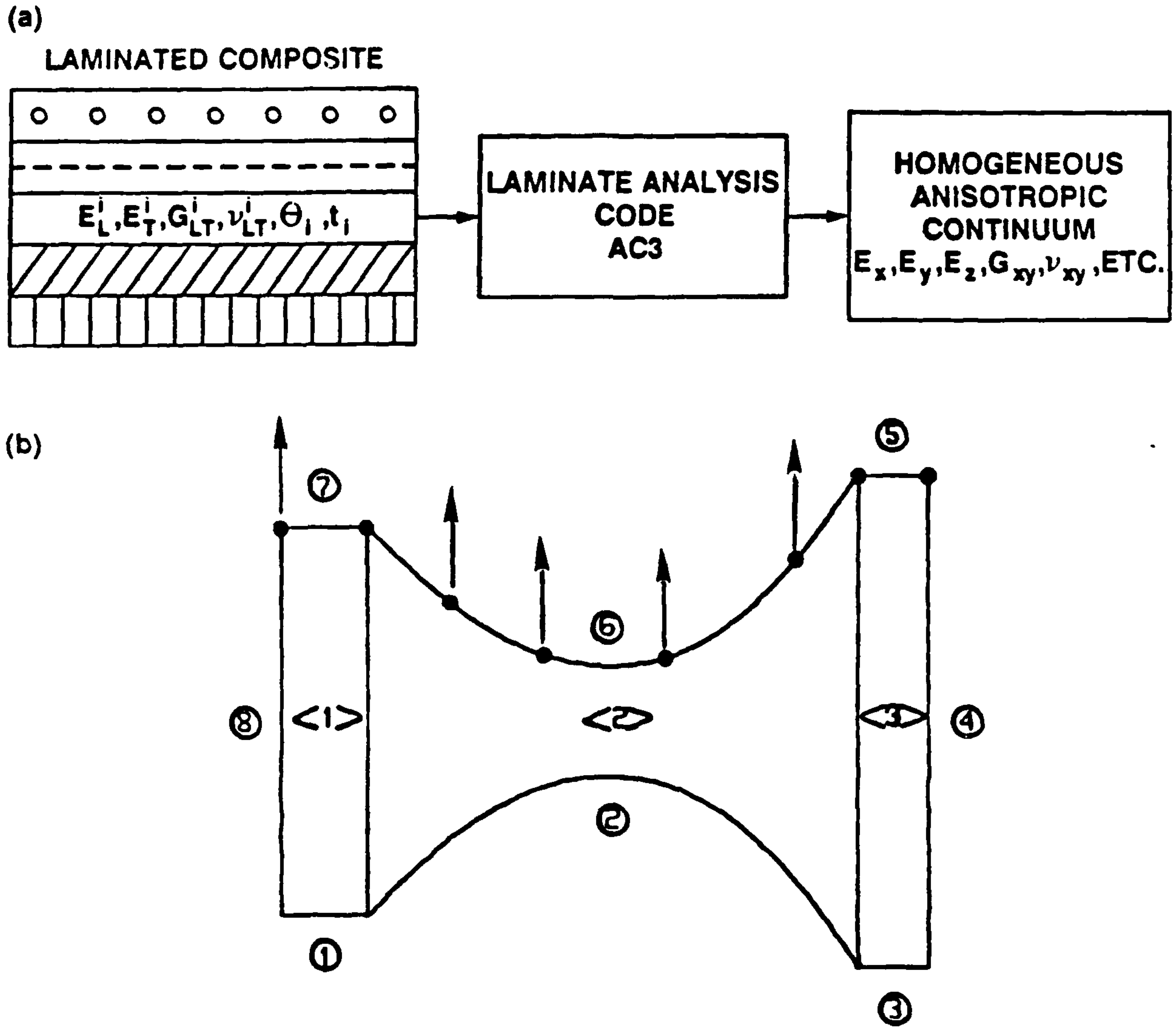
Design Variables: Fiber volume fraction, longitudinal fiber modulus and transverse matrix strength

Objective Function: Specific longitudinal ply strength

Constraints: Thermal expansion coefficient of the ply

Ply impact resistance

Transverse modulus and strength of the ply



- SHAPE DESCRIPTION - LINES, ARCS, SPLINES
- DESIGN VARIABLES - X, Y COORDINATES OF DESIGN POINTS
- DESIGN CONSTRAINTS - TIED TO GEOMETRY

AVERAGE HOOP STRESS ON CURVE (4) ≤ ---

HOOP STRESS IN ZONE <3> ≤ ---

MAXIMUM EFFECTIVE STRESS ON BOUNDARY EXCLUDING (4) ≤ ---

Figure 1 (a) The AC3 laminate analysis code, (b) the geometry-based approach for specifying the optimization problem.

Processing-induced residual stresses for MMC plies.

Relative to software implementation of the above formulation, the numerical optimization code COPES/ADS was integrated with a micromechanical analysis code which was developed as a part of this work and with the MMC residual analysis code RESLAM developed under DICE II – Task 3.4.5.2. This development and example results will be discussed in some detail in the following section.

TECHNICAL RESULTS

Technical accomplishments of the present task can be grouped as follows: (i) methodology development and DESIGN_OPT software enhancements for both shape and ply optimization of 2-D composites, (ii) development of a design model and its software implementation, MICROMECH_OPT, for ply properties optimization using micromechanical design parameters, and (iii) further literature review on composites optimization. These developments and several example results are briefly discussed below.

Shape/Ply Optimization of 2-D Composites

Using the GE CR&D structural optimization software as the basis, a capability was developed for ply thicknesses and angles optimization of 2-D composites under the DICE I project [5]. In the present work, enhancements of this capability were carried out such that both the structural shape and ply geometries can be used as design parameters during the optimization process. A schematic of the software architecture is illustrated in Figure 2. Numerical optimization is carried out using a public domain/commercial computer code COPES/ADS [2,3]. It consists of a variety of commonly employed optimization algorithms including the methods of feasible direction, sequential unconstrained minimization technique and sequential linear programming. A commercial finite element code ANSYS was employed for composite structural analysis. Attention was presently restricted to 2-D

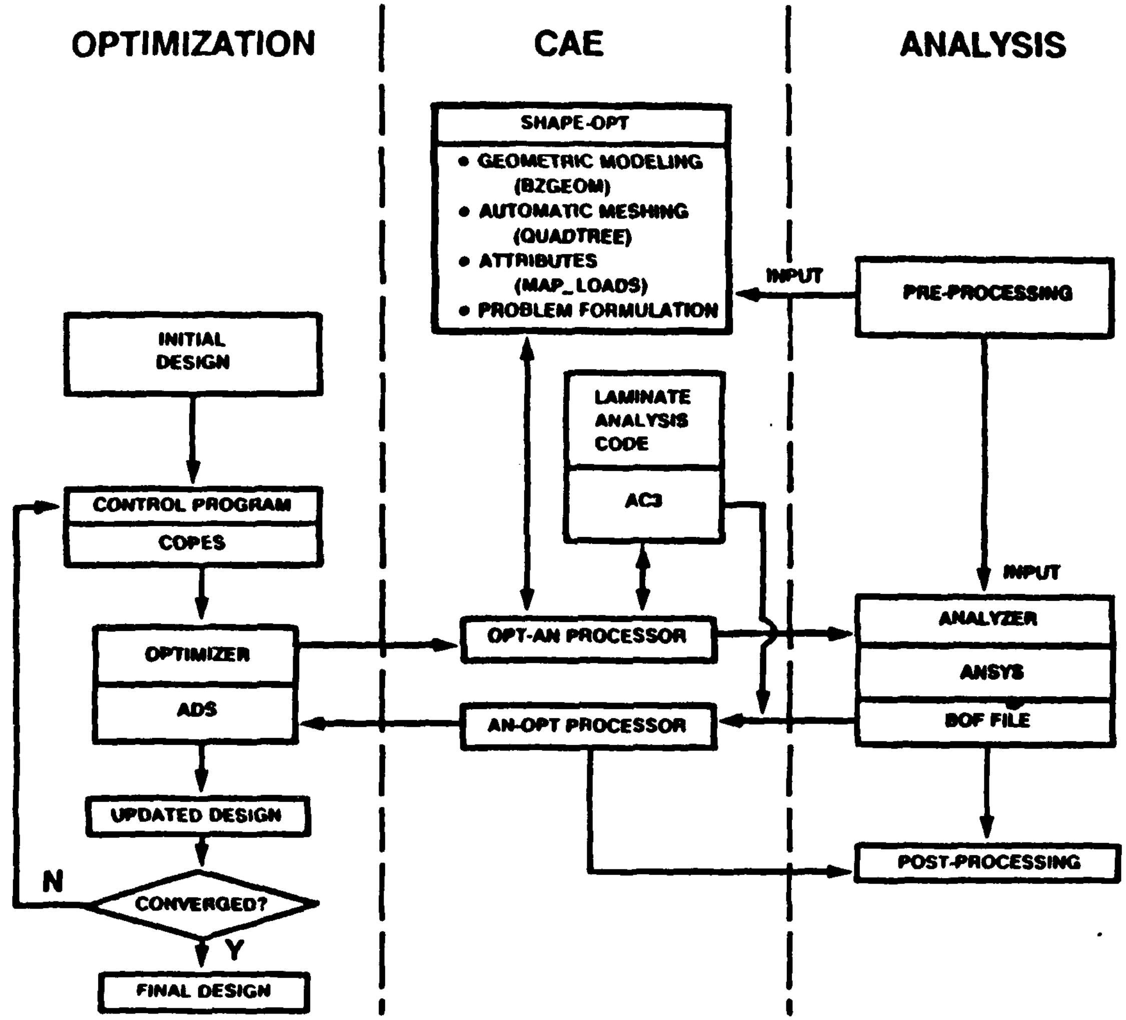


FIGURE 2. Schematic illustration of the DESIGN-OPT software architecture.

components involving plane stress, plane strain or axisymmetric conditions, but both cases of stress and frequency analyses were considered. In Figure 2, OPT-AN and AN-OPT represent software modules for data interchange between the numerical optimization code ADS and the finite element analysis code ANSYS. The OPT-AN module controls the flow of the data from the optimizer to the analyzer, i.e., it maps the design parameters (ply thickness and angles) onto the orthotropic material properties part of the ANSYS input file through an intermediate computer code AC3. OPT_AN also interfaces with another module SHAPE_OPT which essentially integrates various CAE tools like geometric modeling, automatic meshing and geometry-based attribute specification and provides a new finite element mesh and attributes file for the analysis software ANSYS. The significance of SHAPE_OPT also lies in that it allows the user to specify the objective function and constraints at the geometry level rather than at specific finite elements and nodes. The AN_OPT module transmits the structural analysis results of ANSYS to the optimizer through an intermediate universal format binary file BOF. The AN_OPT module and the BOF file are also interfaced with various post-processing software packages (SUPERTAB, MOVIE.BYU, PLOT10) so that the user can obtain an interactive graphics display of the stress contours, finite element model and optimization iteration histories of design variables, objective function and design constraints.

As indicated earlier, the present work was interfaced with the micromechanical modeling effort through a laminate analysis code AC3. It serves two purposes; first, given thicknesses, angles and material properties of various plies and their layout, it computes equivalent orthotropic material constants of a homogeneous medium which are subsequently used by the ANSYS code. Secondly, it calculates allowable strength values which are utilized by the AN_OPT module in determining strength-related design constraints.

We next present results for a number of composites optimization problems that were analyzed as a part of the present work. We first consider the shape and ply optimization of a laminated composite plate subjected to in-plane tensile loading as illustrated in Figures 3a and 3b. Assuming a plane stress condition and the laminate to be balanced and symmetric, and given the ply angles, the optimization problem consists of determining the plate length (a) and width (b) and the ply thicknesses (t_i) which will minimize the plate weight (W) subjected to strength and stiffness constraints. This problem without shape changes was analyzed in DICE I project [5] and also by other researchers [6-8]. The results obtained are shown in Figures 3c and 3d and also presented in Table 1. Since the applied loading is along the x-direction, the optimal design is expected to primarily consist of 0° plies, i.e, plies having fibers along the x-axis. Consequently, we find from Table 1 that the optimal thicknesses of 30°, 60° and 90° plies are very small compared to the 0° plies. The optimal plate dimensions, length and width, are also given in the table; these are found to be reduced by about 30% as compared to initial dimensions. As a result the optimal weight corresponding to the present study with both shape and ply optimization is about 70% smaller than the results for the case in which only the ply thicknesses were considered as design parameters. From Figures 3c and 3d we observe that the finite element mesh was generated automatically as the plate shape (length and width) changed during optimization.

Design optimization of a plane stress plate with a hole subjected to biaxial loading was analyzed as another example (Figures 4a and 4b). The optimization formulation in this case consists of ply thicknesses, plate dimensions and the hole radius (r_c) as the design parameters, minimization of weight as the objective function and stiffness and strength limits as the design constraints. From the results given in Table 2, we find the plate dimensions are reduced by about 10% and the initial hole size by about 50% resulting in about 15% decrease in the plate

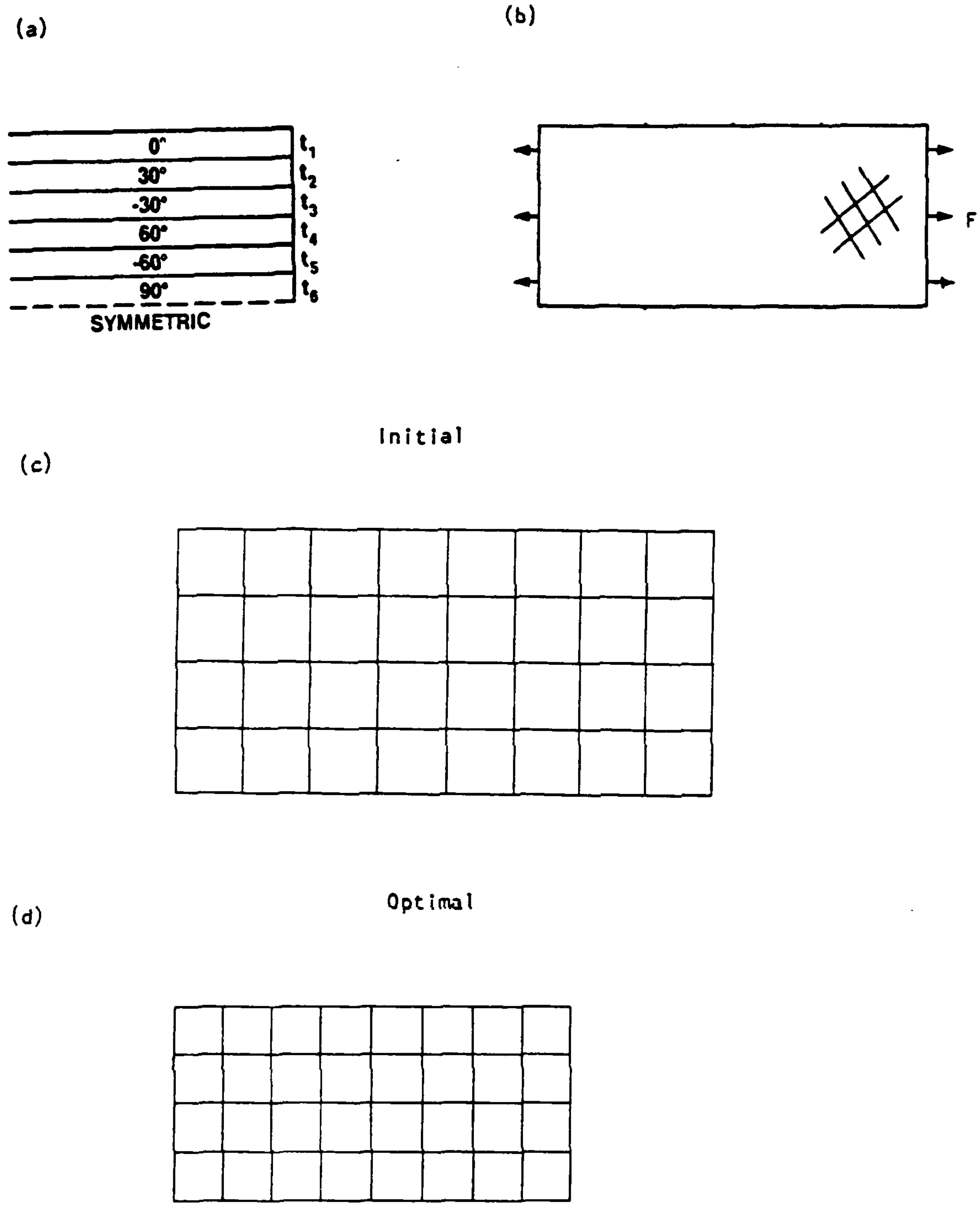


FIGURE 3. Composite plate without a hole under uniaxial load, (a) ply layout, (b) schematic of plate, (c) initial design and (d) optimal design for a quarter of the plate.

TABLE 1. Results from optimization of a composite plate without a hole.

	Initial Values	Optimal Results for Ply-Thickness Optimization [5]	Optimal Results for Ply-Thickness and Shape Optimization
$t_1(0°)$	0.0417	0.100	0.1000
$t_2(30°)$	0.0417	0.0911	0.0351
$t_4(60°)$	0.0417	0.0110	0.0101
$t_6(90°)$	0.0417	0.0163	0.0100
a	10.0	10.0	7.18
b	5.0	5.0	3.59
W	1.44	1.85	0.597

TABLE 2. Results from optimization of a composite plate with a hole.

	Initial Values	Optimal Results for Ply-Thickness Optimization [5]	Optimal Results for Ply-Thickness and Shape Optimization
$t_1(0°)$	0.0417	0.200	0.20
$t_2(30°)$	0.0417	0.200	0.20
$t_4(60°)$	0.0417	0.152	0.165
$t_6(90°)$	0.0417	0.200	0.20
a	10.0	10.0	9.20
b	5.0	5.0	4.50
r_c	1.0	1.0	0.507
W	1.42	6.28	5.38

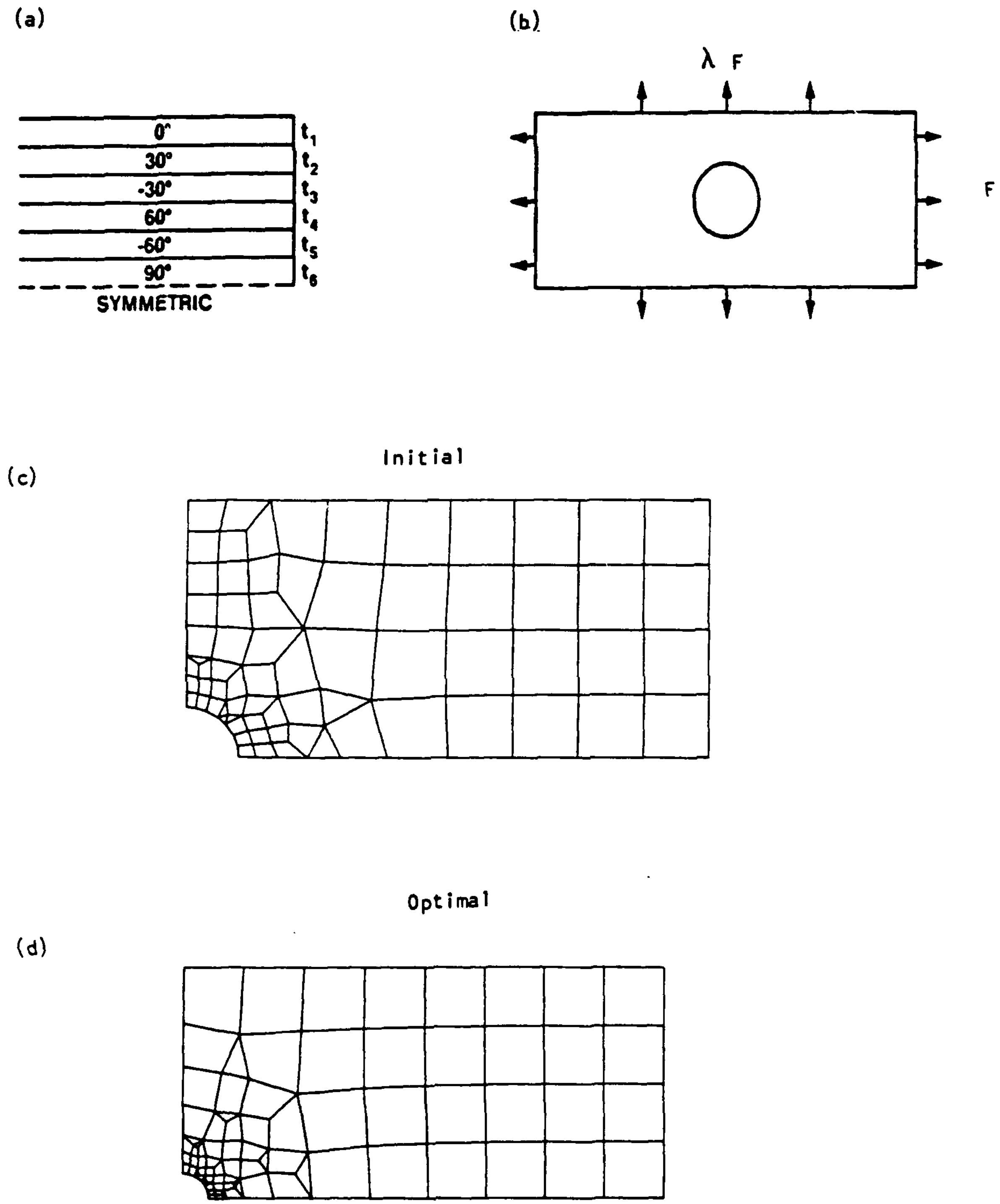


FIGURE 4. Composite plate with a hole under biaxial load, (a) ply layout, (b) schematic of plate, (c) initial design and (d) optimal design.

weight as compared to the ply-only optimization case examined earlier [5]. The initial and optimal plate geometries together with the associated finite element meshes are shown in Figures 4c and 4d. The optimal design with smaller hole radius is found to have more elements near the hole, as expected, than the initial design with a larger hole; this observation clearly reveals the importance and usefulness of automatic mesh generation when both shape and ply are considered as design variables.

The 2-D shape optimization of an axisymmetric composite disc with cross section as shown in Figure 5a is presented next. The optimization problem in this case consists of finding the axisymmetric shape which would minimize the weight of the disc while satisfying constraints on radial, tangential and Von Mises stresses, burst margin and geometric constraints. The design variables are the x and y coordinates of design points as illustrated by the arrows in Figure 5a. The disc is composed of 0° plies with fibers in the hoop direction as shown in the figure. It is subjected to centrifugal loading, and the loading due to the blades is also applied in the form of uniform pressure on the rim. The initial and optimal designs together with the automatically generated finite element meshes are shown in Figures 5b and 5c respectively. The initial design had several violated stress constraints as well as the burst margin constraint. There was a reduction in weight from 14.5 to 12.9 lbs. for the final design with all the constraints satisfied. The next step would be to include ply related parameters, e.g., the fiber volume fraction, as an additional design variable in the optimization formulation; this will be investigated in the DICE III project.

Micromechanical Optimization of Unidirectional Plies

A design model and a computer code MICROMECH_OPT were developed for optimizing properties of unidirectional plies using micromechanical parameters, such as the fiber and matrix properties and the fiber volume fraction, as the design variables. This involved

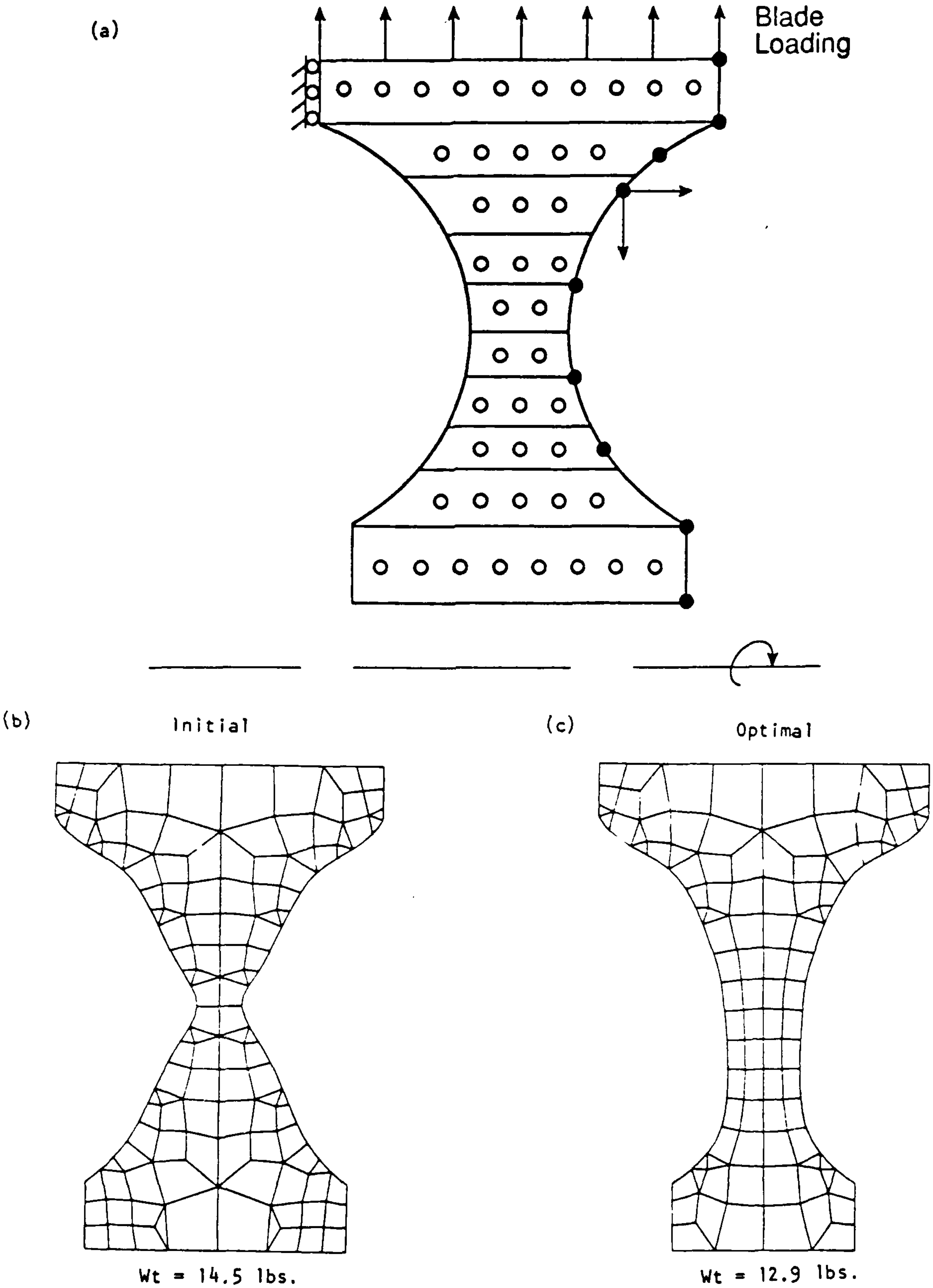


FIGURE 5. Axisymmetric composite disc, (a) schematic of cross section, (b) initial design and (c) optimal design.

integration of the numerical optimization code COPES/ADS with two analysis software related to micromechanical modeling: (i) a computer code PLYPROP which, given the fiber and matrix properties and the fiber volume fraction, computes the ply properties using state-of-the-art micromechanical models, and (ii) a computer model MICROAN which predicts the effect of residual stresses due to the laminating process on the mechanical behavior of MMC's and was developed under the DICE II – Task 3.4.5.2. The PLYPROP code employs the micromechanical relationships developed by Chamis [9] for mechanical, thermal and hygrothermal properties, the model by Rosin and Hashin [10] for transverse coefficient of thermal expansion, in-plane strength theory by Tsai [11], and Chamis [12] equations for fracture toughness, impact resistance and through-the-thickness strength properties. These models were chosen because of their correlation against 3-D finite element analysis or experimental data as reported in the literature[13,14]. The MICROAN model by Nimmer [DICE II – Task 3.4.5.2.] aims at predicting the influence of fiber volume fraction and fibers spacing ratio on the micromechanical stress state, fiber-matrix separation, matrix yielding and fiber failure under the influence of applied mechanical loading and thermal gradient relative to the processing temperature.

We next present the MICROMECH_OPT results for property optimization of graphite/epoxy unidirectional plies. The first case considered involves maximization of specific longitudinal modulus $\sqrt{E_{11}}/\rho$ of a ply by using fiber volume fraction, fiber and matrix densities and fiber longitudinal modulus as the design parameters. From the results presented in Table 3, we find that optimization resulted in an increase of $\sqrt{E_{11}}/\rho$ from 7.28 to 12.17. In the second case, optimization was carried out by placing constraints on longitudinal thermal coefficient of expansion and impact-resistance properties. Referring to results in Table 3, the optimal value of $\sqrt{E_{11}}/\rho$ is about 10% smaller than the previous unconstrained

TABLE 3. Optimization results for graphite/epoxy uniplies. Maximization of $\sqrt{E_{11}}/\rho$ for buckling applications.

		Initial Value	Lower Bound	Upper Bound	Optimal Value
Case 1					
Maximize $\sqrt{E_{11}}/\rho$	$\left(\frac{\sqrt{GPa}}{g/cm^3}\right)$	7.28	-	-	12.17
Design Variables					
V_f		0.62	0.45	0.70	0.70
ρ_f	(g/cm^3)	1.80	1.47	2.63	1.47
ρ_m	(g/cm^3)	1.38	1.16	1.38	1.16
E_{f11}	(GPa)	228	85	400	400
Case 2					
Maximize $\sqrt{E_{11}}/\rho$	$\left(\frac{\sqrt{GPa}}{g/cm^3}\right)$	7.28	-	-	10.92
Design Variables					
V_f		0.62	0.45	0.70	0.678
ρ_f	(g/cm^3)	1.80	1.47	2.63	1.47
ρ_m	(g/cm^3)	1.38	1.16	1.38	1.16
E_{f11}	(GPa)	228	85	400	329
Constraints					
α_{11}	$(\mu m/m/°C)$	-0.602	-1.0	1.0	-0.781
IR^*_{23S} (interlaminar shear)	(MPa)	0.951	0.9	-	0.899
IR_{11F} (longitudinal flexural)	(MPa)	15.8	12.0	-	12.0
IR_{22F} (transverse flexural)	(MPa)	0.632	0.5	-	0.612
Case 3					
Maximize $\sqrt{E_{11}}/\rho$	$\left(\frac{\sqrt{GPa}}{g/cm^3}\right)$	32.9	-	-	12.19
Design Variables					
V_f		0.62	0.45	0.70	0.70
ρ_f	(g/cm^3)	1.80	1.47	2.63	1.47
ρ_m	(g/cm^3)	1.38	1.16	1.38	1.16
E_{f11}	(GPa)	228	85	400	400
E_m		3.45	2.21	5.17	5.17
S_{ft}		3105	1724	4137	3203
S_{fc}		1793	350	4826	2014
Constraints					
α_{11}	$(\mu m/m/°C)$	-0.602	-1.0	1.0	-0.758
IR_{23S} (interlaminar shear)	(MPa)	0.951	0.9	-	0.900
IR_{11F} (longitudinal flexural)	(MPa)	15.8	12.0	-	12.0
IR_{22F} (transverse flexural)	(MPa)	0.632	0.5	-	0.537

*IR ≡ Impact Energy Density

case, but the final values of the properties constraints are within the imposed limits. The final case includes three additional design variables (matrix Young's modulus and fiber strengths in tension and compression) in the optimization process. Because of the influence of these parameters on impact resistance, the design objective $\sqrt{E_{11}}/\rho$ is found to improve significantly while maintaining the impact resistance constraints satisfied. In summary, this numerical study has shown that ply properties can be optimized, at least mathematically, by adjusting properties of the constituent materials.

Specific longitudinal and transverse strengths, S_{11T}/ρ and S_{22T}/ρ, were chosen as design objectives in a subsequent set of optimization studies. From the results presented in Table 4, it is observed that the specific transverse strength increased by about 25% whereas a 75% increase in the longitudinal property was achieved. Also, notice that not all the design variables changed to their lower or upper bounds. Certain property limits that were imposed as optimization constraints were found to remain satisfied during the optimization iterations. As a final note, the property improvements as reported here may not be physically attainable with existing materials, but the model developed herein can be useful for providing guidance in tailoring properties of new composite material systems.

Finally, we consider the MICROMECH_OPT results for metal matrix plies made from SiC fibers and a Ti-based matrix. In the example considered, we wish to maximize the longitudinal modulus of the ply by using fiber volume fraction as the design parameter. The optimization results are presented in Table 5 for an applied longitudinal stress of 700 MPa at a temperature of 927°C below the processing temperature. In the unconstrained case, the fiber volume fraction increased from 0.35 to 0.40 resulting in an increase of the modulus from 208 GPa to 224 GPa. However, the matrix effective stress exceeds the yield stress of the matrix (σ_y = 700 MPa). The optimization model was run again with constraints imposed

TABLE 4. Optimization results for graphite/epoxy uniplies. Maximization of specific strengths.

		Initial Value	Lower Bound	Upper Bound	Optimal Value
Case 1					
Maximize S_{11}/ρ	$\left(\frac{MPa}{g/cm^3}\right)$	1013	-	-	1815
Design Variables					
V_f		0.62	0.45	0.70	0.70
ρ_f	(g/cm^3)	1.80	1.47	2.63	1.47
ρ_m	(g/cm^3)	1.38	1.16	1.38	1.16
S_{ft} (fiber tensile strength)	(MPa)	3105	1724	4137	4137
Case 2					
Maximize S_{11}/ρ	$\left(\frac{\text{MPa}}{g/cm^3}\right)$	1013	-	-	1815
Design Variables					
Same as Case 1 plus					
G_{f23}	(GPa)	6.89	1.52	167	6.87
S_{ms} (matrix shear strength)	(MPa)	101	55	483	102
Constraint					
IR_{23S} (interlaminar shear)	(MPa)	0.951	0.9	-	0.902
Case 3					
Maximize S_{22}/ρ	$\left(\frac{\text{MPa}}{g/cm^3}\right)$	32.9	-	-	41.5
Design Variables					
V_f		0.62	0.45	0.70	0.45
ρ_f	(g/cm^3)	1.80	1.47	2.63	1.47
ρ_m	(g/cm^3)	1.38	1.16	1.38	1.16
Case 4					
Maximize S_{22}/ρ	$\left(\frac{MPa}{g/cm^3}\right)$	32.9	-	-	41.5
Design Variables					
Same as Case 3 plus					
E_{f22}	(GPa)	13.8	4.1	400	15.6
E_m	(GPa)	3.45	2.21	5.17	4.04
G_{f23}	(GPa)	6.89	1.52	167	6.97
α_m	$(\mu m/m/°C)$	41.2	36.0	103	41.8
S_{ft} (fiber tensile strength)	(MPa)	3105	1724	4137	4137
Constraints					
E_{22}	(GPa)	8.43	8.00	-	8.02
S_{11T}(ply long tensile strength)	(MPa)	1661	1600	-	1606
α_{22}	$(\mu m/m°C)$	48.2	-50	50	49.9

TABLE 5. Optimization results for SiC/Ti-based metal matrix composites. Maximization of E_{11}.

	Initial Value	Lower Bound	Upper Bound	Optimal Value
Case 1				
Maximize E_{11} (GPa)	208	-	-	224
Design Variable				
V_f	0.35	0.25	0.40	0.40
Microstresses (MPa)				
σ_{eff} (matrix elements A&C)	733	-	-	772
σ_{eff} (matrix element B)	412	-	-	432
σ_{22} (fiber)	-385	-	-	-350
Case 2				
Maximize E_{11} (GPa)	208	-	-	196
Design Variables				
V_f	0.35	0.25	0.40	0.31
Microstresses (MPa)				
σ_{eff} (matrix elements A&C)	733	-700	700	700
σ_{eff} (matrix element B)	412	-700	700	394
σ_{22} (fiber)	-385	-3447	0	-410

such that the matrix effective stress is not exceeded. Additionally, a constraint on the fiber transverse stress was imposed such that the stress is always compressive in order to avoid fiber/matrix debonding. This resulted in the fiber volume fraction decreasing from 0.35 to 0.312 and the modulus decreasing from 208 GPa to 196 GPa. Note that the optimal result for the modulus is smaller than the initial value due to violation of the effective stress constraint associated with the initial design values. As a final remark, it is pointed out that this example incorporates the effect of process induced residual stresses on the subsequent mechanical response under the action of an operating load.

Literature Review on Composites Optimization

In the DICE I project, an effort was started to perform a literature review on composites optimization and to develop some generic optimization formulations in terms of design variables, objective function(s) or performance measure(s) and design constraints. This work was further continued in the present project, and several other papers and abstracts were compiled and reviewed. Optimization formulations that have been developed will soon be documented in the form of a technical report.

CONCLUSIONS

A number of conclusions have been reached from the results described in the previous section. First, both ply and shape optimization was found to give better optimal results than those obtained from ply optimization alone as reported in DICE I report [5]. Although not discussed here, it was found that the final result does depend upon whether size or shape optimization is performed first and also upon whether size and shape optimization is carried out simultaneously. This issue will be further investigated in the DICE III program. From the micromechanical ply optimization results, it can be concluded that the optimal response and design of a composite structural component may require optimization with respect to

micromechanical parameters. An example of this would be the shape optimization of a disc with the fiber volume fraction as the micromechanical design parameter and the associated microlevel stress constraints. Finally, our composites optimization work has thus far been based upon an approach which transforms the laminated composite into an homogeneous anisotropic continuum through the use of a point stress analysis program. An alternate approach for plane stress and shell type composites would be to use a laminated composite shell element. This is being investigated in the DICE III program.

RECOMMENDATIONS

A software capability has been developed for shape and ply optimization of 2D (plane stress, plane strain and axisymmetric) composite structural components. An important enhancement would be shape/ply optimization of composite plates, shells and solids. Also, it would be desirable from both technical and practical viewpoints to incorporate the micromechanical optimization concepts into shape/ply optimization methodology so that the structural shape, ply thicknesses, ply angles and micromechanical parameters (e.g., fiber volume fraction) can be used as design variables to achieve the optimal composite design and the associated structural response. Several of these issues are being addressed in the DICE III project.

PUBLICATIONS

None

SOFTWARE LIST

1. COPES/ADS – Commercial software package available from Engineering Design Optimization, Inc., Santa Barbara, CA.

2. ANSYS – Commercial finite element code available from Swanson

Analysis Systems, Pittsburgh, PA.

3. SHAPE_OPT – A software module integrating various CAE codes, developed under GE funds.

4. DESIGN_OPT – A structural optimization software developed under GE funds.

5. COMP_OPT – Composites related enhancements of DESIGN_OPT developed under DICE funds.

6. MICROMECH_OPT – Micromechanical ply optimization software developed under DICE funds.

REFERENCES

[1] Vanderplaats, G. N., *Numerical Optimization Techniques for Engineering Design: With Applications,* McGraw-Hill, 1984.

[2] Vanderplaats, G. N. and Sugimoto, H., "A General Purpose Optimization Program for Engineering Design," Int. J. Comp. Struct., Vol. 24, No. 1, 1986.

[3] Vanderplaats, G. N., "COPES/ADS - A FORTRAN Control Program for Engineering Synthesis Using the ADS Optimization Program: User's Manual," Engineering Design Optimization, Inc., June 1985.

[4] Kumar, V., German, M. D. and Lee, S.-J., "A Geometry-Based 2-Dimensional Shape Optimization Methodology and a Software System with Applications," Third Int. CARS/FOF Conf. Proceed., Southfield, MI, Aug. 1988.

[5] Kumar, V., German, M. D., Lee, Y-J., Nimmer, R. P., "Optimization of Composites," DICE I – (Task 3.4.9.2), Final Report.

[6] Schmit, L. A., Jr., and Farshi, B., "Optimum Laminate Design for Strength and Stiffness," International Journal for Numerical Methods in Engineering, Vol. 7, 1973, pp. 519-536.

[7] Schmit, L. A., Jr., and Farshi, B., "Optimum Design of Laminated Fibre Composite Plates," International Journal for Numerical Methods, Vol. 11, 1977,m pp. 623-640.

[8] Dhir, S. K., "Optimization of Openings in Plates Under Plane Stress," AIAA Journal, Vol. 21, No. 10, Oct. 1983, pp. 1444-1447.

[9] Chamis, C. C., "Simplified Composite Micromechanics Equations of Hygral, Thermal and Mechanical Properties," SAMPE Quarterly, Vol. 15, No. 3, April 1984, pp. 14-23.

[10] Rosen, B. W. and Hashin, Z., "Effective Thermal Expansion Coefficients and Specific Heats of Composite Materials," International Journal Engineering Science, Vol. 8, 1970, pp. 157-173.

[11] Tsai, S. W., *Composites Design*, 4th Edition, Think Composites, Dayton, 1988.

[12] Chamis, C. C., "Simplified Composite Micromechanics Equations for Strength, Fracture Toughness, and Environmental Effects," SAMPE Quarterly, Vol. 15, No. 4, July 1984, pp. 41-55.

[13] Caruso, J. J. and Chamis, C. C., "Assessment of Simplified Composite Micromechanics Using Three-Dimensional Finite-Element Analysis," Journal of Composites Technology & Research, Vol. 8, No. 3, Fall 1986, pp. 77-83.

[14] Bowles, D. E. and Tompkins, S. S., "Prediction of Coefficients of Thermal Expansion for Unidirectional Composites," Journal of Composite Materials, Vol. 23, April 1989, pp. 370-388.

DICE PROGRAM

Task 4.5.5 Unified Life Cycle Engineering **Stanford University**

Objectives:

1. Introduction and Phase-II Research Objectives

The Research focused on two distinct issues in concurrent engineering: one is an engineering issue and the other is a management issue. A salient characteristic of concurrent engineering is the desire to allow a "later" subprocess to begin before an "earlier" subprocess is terminated.[1] For example, a product team may begin to design before knowing the full specifications and realm of possibility. With a high degree of concurrency, the information gained by actually beginning later subprocesses may help to reduce critical uncertainties in the decisions at earlier stages and guide participants to undertake more productive and efficient courses of action.

It is critical that participants in the different subprocesses be able to communicate with each other in a timely and efficient manner, knowing quickly how the decisions and actions of certain participants impact other participants' choices. This technical requirement can be solved via development of an integrated computing and communications environment that supports concurrent development activities. This is the main focus of many engineering research efforts in concurrent engineering. This research focused primarily on developing the means to help participants in a concurrent-engineering enterprise efficiently coordinate the technical-engineering activities that are carried out in various stages of this process. For example, a good deal of this research focuses on issues such as design for manufacturability, evolution of product design along with development of specifications and requirements and the computational and communications technology that supports these types of activities.

Another central feature of concurrent engineering creates important management challenges. In an "overlapping" approach of concurrent engineering, the end of each subprocess may not be well defined. In fact,

[1] A subprocess can be said to be "later" in the sense that it depends upon decisions or outcome produced by other "earlier" processes.

because of the dependence on concurrency, the "official" ending of each subprocess may not be at all crucial. As long as conflicts can be resolved when they arise, one can adopt the view that all subprocesses end at the same time. However, the intensity of activity during completion of each subprocess may have a varying profile with multiple "peaks" during the whole period. For such an approach, the coordination of these activity profiles in all the subprocesses is a major management issue. The difficulty in coordinating these activities arises from the unique characteristics of the uncertainty resulting as a consequence of the overlapping process.

In an overlapping process, the participants in some subprocess may need to cope with a great degree of ambiguity in the information upon which their early problem-solving activities depend since "upstream" subprocesses may not have yet specified requirements on this subprocess. However, as time and the upstream processes progress, ambiguity reduces. While a great deal of ambiguity exists, the team members need to provide for flexibility in their activities and commitments, such as investigating alternative paths and building in contingencies to cope with change and unexpected events. When ambiguity is reduced to some threshold level, problem solving activities in a subprocess can then focus on the best, specific course of action. In summary, managing concurrent engineering requires assessing the degree of ambiguity that exists as the various subprocesses unfold and then determining the "optimum" level of flexibility and contingency planning to provide throughout the development process.

The Phase II research in this area focused on these issues of technical integration among the engineering subprocesses and, in particular, maintenance of maximum desired flexibility throughout the product development process. The Phase II effort aimed to:

- select and analyze an illustrative product-development process with respect to the issues just introduced;

- formulate a mathematically grounded approach to coping with ambiguity in the development cycle by managing the degree of flexibility in the way in which subprocesses are carried out;

- formulate a concept of decision support that embodies and aids participants in the concurrent engineering process in enacting the preceding management approach and
- specify a computational architecture that will deliver this decision-support methodology and integrate into the DICE computational environment.

2. Analysis of Turbine-Blade Design Problem

A specific example of a turbine blade design process is used as a case to guide an analytical formulation and a mathematical solution method for the problem. In this design process, the major issues are the selection of material used as well as the choice in process technology. The problem is divided into two-levels -- a "lower level problem" formulated by applying an extended form of Taguchi's method[2] to design specification and an "upper level problem".

2.1 Lower Level Problem

The assumption is that a specific material is used and a process technology is chosen. A parametric optimal quality design problem is formulated similar to the Taguchi method with product specification as a parameter. The "best" design specification is then formulated as another optimization problem with the Taguchi method nested within it. This provides a methodology for integrating the Taguchi method with design specification.

2.2 Upper Level Problem

After the lower level problem for each potential material to be used and possible process technologies is solved, a higher level choice problem is formulated in terms of the set of lower level solutions. If there is no R&D uncertainty, the upper level problem is reduced to one of cost/benefit analysis. Whereas if there is R&D uncertainty, an optimum trade-off curve between prototype development cost and probability of successful design is introduced to represent the measure of flexibility. Based on this measure, a methodology for determining the optimum level of flexibility can be developed.

[2]Taguchi, G.and Wu, Y., "Introduction to Off-Line Quality Control", Central Japan Quality Control Association, Japan, 1980.

Approach:

3. Mathematical Solution Approach

3.1 Lower Level Problem

A mathematical optimization method is developed to solve the Lower Level Problem. The method integrates the Taguchi method with design specification via a two-level hierarchical optimization problem.

3.2 Upper Level Problem

For the Upper Level Problem, an optimization solution method is developed for the case where there is no R&D uncertainty. When R&D uncertainty is present, a solution algorithm is developed based on dynamic programming to derive the optimum trade-off curve between prototype development cost and probability of successful design. The trade-off curve provides the basic framework to define the notion of flexibility and the fundamental for determining the optimum level of flexibility required in the early design process where R&D activities are to be carried out concurrently with product design. In determining the optimum flexibility level budgetary constraint, human resource constraint, competitive pressure and status of technology development can be incorporated in the basic framework to result in a man-machine interactive solution methodology. A simple example is used to illustrate the working of the solution method and its implication in managing flexibility.

Results:

4. Decision Support Approach

The research team is investigating how the mathematical methods described above can be delivered as part of a management decision-support "tool" that will assist concurrent engineering managers and other participants in exploiting the right degree of flexibility. The approach formulated during Phase II takes the preceding mathematical analyses as its core and augments them with additional facilities for planning and scheduling of the manpower and other

resources needed to carry out prototyping activities. The principal effort during Phase II focused on the formulation of a computational architecture and user-support environment to be used in further development of this decision-support approach during Phase III.

4.1 Resource Planning

Using the mathematical methods described in the previous section, a manager is presented with a set of one or more feasible prototyping options that commit the organization to an effort involving one or more concurrent prototyping efforts, each exploring certain options for product design and production process. In applying these mathematical methods, the user (or some other source of information) provides estimates of the time to be spent in prototyping and bounds on resources to be expended. These estimates can be quite non-specific, and more importantly, they can embody a great deal of uncertainty.

In this phase, some capabilities needed by a decision-support environment in order to enhance the usefulness of the preceding analytic methods were explored. Moreover, some illustrative cases that typify the process of turbine blade design were formulated. Based on this preliminary examination, we formulated the following set of features to be embodied in the decision-support system to be developed during the next phase of this research:

- Manpower allocation tools

  The decision-support environment will offer tools for accessing available data-bases containing personnel information and making provisional and final staffing commitments as required by the planning options generated by the analytic methodology described above. This set of tools should also provide facilities to further refine the costs and other associated resource expenditures;

- Project planning tools

  Again using the information from the analytic methodology as a starting point, the system will offer a variety of support tools to assist the manager in laying out detailed milestones and other plans for implementing the available options. At the least, this system should provide the ability to

construct milestone charts, PERT charts, GANTT charts describing project actions. To construct these plan descriptions the manager may require access to data that resides in other elements of the DICE architecture. The plans being constructed should describe any coordination or synchronization required between the prototyping activities and other activities that may be ongoing in the concurrent engineering process. Finally, these plan description tools should be augmented with tools that assist a manager in constructing a more precise plan; and

- Option refinement support
  As detailed plans are constructed, the manager may determine that his original estimates of cost constraints should be modified. In this case, this support environment should allow the manager to rerun the analytic methodology using the new estimates. In this way, the planning and resource-allocation tools work with the analytic methods developed to support iterative development and refinement of prototyping options.

4.2 Computational Architecture

During Phase II, an architecture was refined and specified to deliver the analytic methods and the planning-support tools. The specified architecture is based on the Schemer problem-solving architecture (Fehling et al., 1989). Schemer is a general-purpose problem solving architecture specially formulated to support real-time performance and implementation of distributed, concurrent problem-solving systems. Since both real-time performance and distributed, concurrent operation are fundamental aspects of the DICE system, Schemer provides an excellent starting point for the work on Phase II of this project.

Figure 1 depicts the main features of Schemer, as a computational architecture for our decision-support environment. Briefly, the figure emphasizes the following important features to be provided with this system:

- Event-directed *Handlers* can be used to monitor for the occurrence of, or change in, information that is critical to the analyses and plan-construction activities being carried out in the decision-support environment;

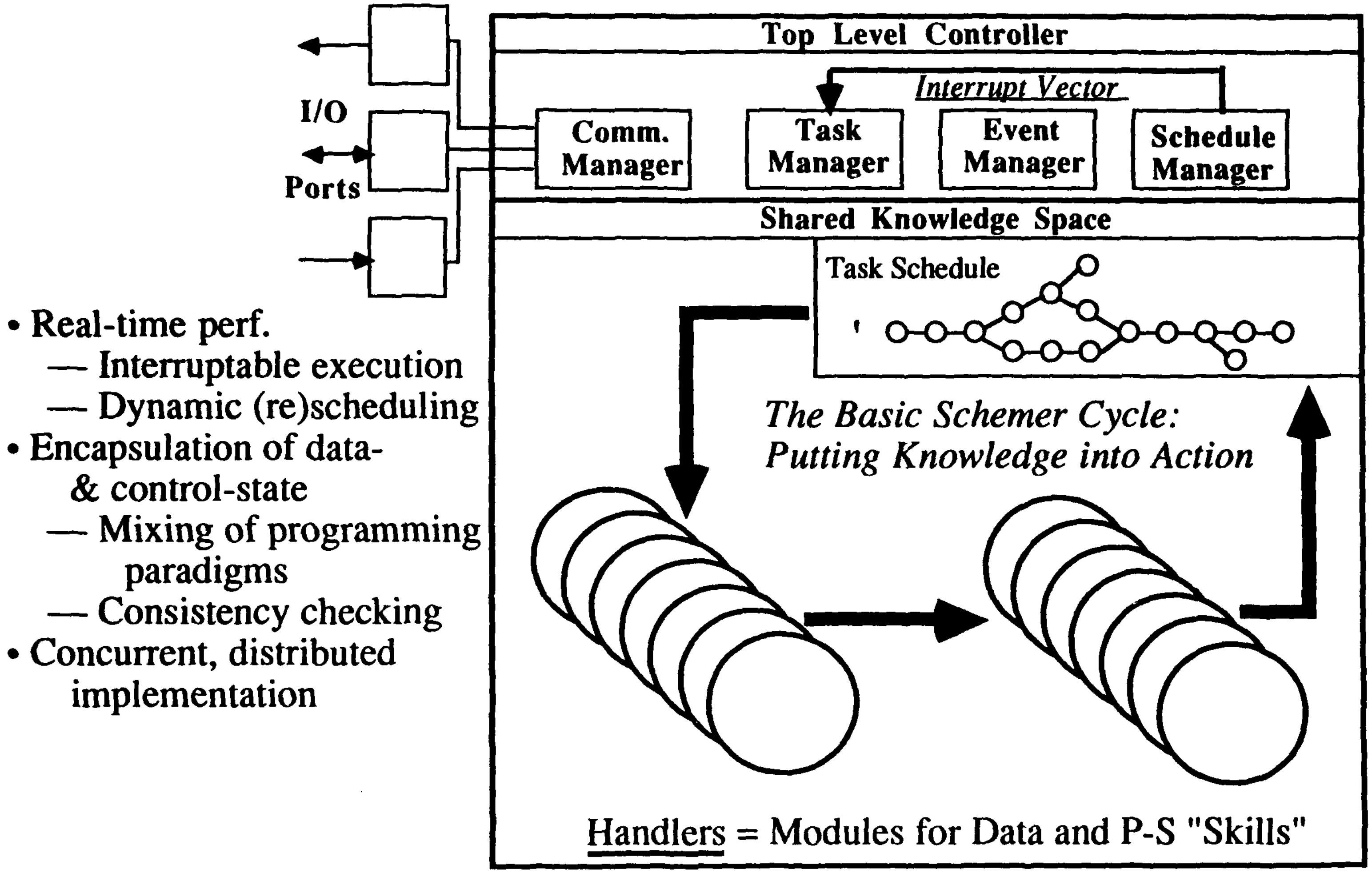


Figure 1 — Schemer Architecture

- *Handlers* can also be used to encapsulate local copies of critical data, providing transaction contexts with which to manage the consistency of data that may be simultaneously accessed by multiple users of this decision-support system and users of other DICE facilities;

- *Communication Ports* can be defined dynamically as a means of creating situation-specific data communication between this system and other elements in the DICE architecture. Such a dynamically defined communication facility provides the basis to establish efficient connections between this system and the loci of large bodies of critically needed information and

- A *Top-Level-Controller* that provides interruptable, event-directed scheduling of the activities of the decision-support system and actions that it is taking on the user's behalf.

Conclusions and Recommendations:

The formulation and solution method developed in this phase of the research provides a fundamental framework to allow an analysis of the notion of flexibility in the design development process as well as providing guidance for management coordination and control on the concurrent R&D and design activities. The solution method also points to the kind of information and modeling that is needed to allow optimum choice of flexibility level. A computational decision-support system is being formulated that uses the the optimum trade-off curve to provide guidance to the project management of concurrent R&D and design. Further research is needed, however, to develop methods for evaluating certain probability functions used in the solution method for constructing an optimum trade-off curve. As our preceding discussion indicates, an assessment methodology that is adaptive in nature — i.e., *a priori* assessment based on extrapolation of past experiences -- may be adaptively updated as prototyping experiences accumulate.

Another research direction to be undertaken in Phase III is to refine our specification of a decision support system to provide the coordination and control of the concurrent R&D and design process. The main issue is how to

maintain the right level of flexibility when there are R&D uncertainties and when to terminate additional R&D influence in product design. The focus will be on construction of a proof of concept prototype that demonstrates the concepts developed in the present phase of this project. This prototype will illustrate the central role played by our methodology for analyzing the optimum trade-off between the prototyping cost and the probability of a successful design.

Publications:

Fehling, M.R. "New Approaches to Computer-Integrated Manufacturing." Special Colloquium presented at the Systems Research Center, University of Maryland, College Park, Maryland, March, 1990.

Fehling, M.R., Altman, A., and Wilber, B.M. "The Heuristic Control Virtual Machine: An Instance of the Schemer Architecture for Real-time Problem-solving." In V. Jagannathan, R. Dodhiawala, and L. Baum (Eds.) Blackboard Architectures and Applications. New York, Academic Press, 1989.

Tse, E. "Concurrent R&D and the Product Design Process." Laboratory for Intelligent Systems Technical Report No. ISL-TR-90-11, February, 1990

Hardware:

None.

DEMONSTRATIONS OF CONCURRENT ENGINEERING

DICE Program

Task 4.5.1 Design Task Management **North Carolina State University**

Objectives:

Traditional methodologies of product development are based on a sequential flow from specifications to a detailed design which is manufactured, tested and delivered to the customer. In reality, this almost never happens. For example, a product specification may progress to the production phase before it is determined that it is not manufacturable or that manufacturing costs for the product as designed are prohibitive. In the worst scenario, the project goes back to the R\&D phase. Such long feedbacks, result in long product development times.

An alternative approach to these traditional methods described above is to take a more fine-grained view of the operations and interactions required to progress from product concept to delivery. The process is represented as a directed graph, in which the nodes of the graph represent primitive operations such as {em evaluate the thermal properties of material Z}, or (design widget A), or (inspect assembly B). The edges coming into a node then represent the information required to perform the operation -- for example, the (functional specification for assembly B). This approach elucidates the explicit dependencies among all of the operaticns required in the product development cycle. The problem then becomes one of mapping the nodes of the graph onto the available resources (people and machines) and scheduling the operations assigned to each resource so as to achieve maximum concurrency. The product development cycle is thus minimized. This approach to product development is one aspect of the current DARPA Initiative in Concurrent Engineering (DICE) program.

When maximizing the concurrency of the design process within given cost constraints, the following questions must be answered:

1. What types and amounts of resources will be needed to complete the design on time?;

2. How will the workers be organized to minimize communication delays and errors?;

3. To whom will the tasks be assigned?, and

4. In what order wili the tasks be performed?

The DICE scheduling problem is difficult to solve due to its combinatorial nature -- i.e., the quality of a solution is affected by a large number (possibly millions) of interacting decisions.

Expressed mathematically, we must find a vector $ (bf s) = (s_1, s_2, cdots, s_{N})$ which minimizes some function of interest, $H({bf s})$, that depends on $bf s$ in some complex, non-linear way. Compounding the difficulty, the problem is (discrete) in that the decisions may assume only one of a limited number of values (e.g., $s_i in \{0, 1}, forall i$).

Typically, the decisions are made based upon a combination of prior experience and guesswork. Unfortunately, experience can bias a schedule away from an original and advantageous configuration and a single poor guess can distort an entire system due to interactions between decisions.

Approach:

At North Carolina State University, the use of simulated annealing and neural networks for solving optimization problems without bias is being examined. Simulated annealing is a gradient descent technique incorporating a random process which allows the escape from local minima such that the globally optimum solution can be found.

Neural networks contain many simple processing elements which are highly interconnected to rapidly solve large problems in a cooperative manner.

The randomness of simulated annealing is incorporated into neural networks to create the Mean Field Annealing (\mfa) algorithm. The controlled randomness improves the solutions found by the neural network while the cooperative and continuous nature of the network increases the speed and parallelism of the annealing process. Thus, mfa can rapidly find near-optimal solutions to a wide variety of problems. Mfa solves scheduling problems by manipulating the

following variables:

s_{ijk} = left { begin{array}{cl}
1 & mbox{if task i is executed on resource j at time k}
0 & mbox{otherwise}
{array}. so as to arrive at a near-optimal schedule.

The variables are updated according to a normalized Boltzmann distribution as follows:

s_{ijk} = frac{exp(-\Phi_{ijk}/T)}{\sum_{l,m}\exp(-\Phi_{ilm}/T)}
where Φ_{ijk} is merely the cost incurred by executing task i on resource j at time k (i.e. $s_{ijk} = 1$).

Lowering the control parameter T (often called the {temperature}) forces each task to converge to the resource and time slot having the lowest overall cost. This cost is dependent upon many of the other task assignments, thus the tasks cooperate and compete for desirable resources and time slots and eventually reach a near-optimal global solution.

Technical Results:

/pals/ was modified and repackaged to form mftp -- the Mean Field Task Planner -- which optimizes task schedules in the DICE environment. Mftp is a tool to be used by the DICE Project Lead (PL) to plan and evaluate the distribution and scheduling of task assignments over the available organizational resources. The PL interfaces to mftp via xs --- an X11-based spreadsheet.

A software tool, ganttview, was also created to allow the optimized schedule derived by mftp to be graphically displayed as a Gantt chart in a workstation window.

In May, the function of mftp was successfully demonstrated on a small problem as part of the "executable mockup" at the dedication of CERC in Morgantown, West Virginia. In solving the demonstration problem, it was found that the execution time is polynomial in the product of the size of the task graph, the number of available organizational resources and the length of the scheduling time horizon. This creates a significant problem, particularly when there is an

attempt to scale up to real problems of interest in the context of DICE. Therefore, a new version of mftp was created which progressively halves the scheduling horizon and determines in which half a given task will reside. For a horizon of T time units, the new algorithm requires $log_2(T)$ iterations to achieve the same precision as the original version of mftp.

However, the execution time of each iteration is now a polynomial of only the size of the task graph and the number of available organizational resources. This led to an overall improvement which allows mftp to be used to interactively on more realistic problems.

The divide-and-conquer approach to scheduling tasks over a time horizon of length T reduced the execution time of mftp by a factor of $log_2(T)/T^2$.

By applying clustering techniques to the tasks and resources, similar reductions were gained in problem dimensionality with a resulting increase in execution speed.The decrease in dimensionality was achieved by clustering sequential or near-sequential tasks with similar resource usage characteristics into {super-tasks} (sts).

These sts were then assigned for execution on a given resource at a specific initiation time. Such clustering can be done using Kohonen neural networks, competitive learning networks or standard hierarchical clustering techniques. As a result of this work, two new clustering techniques were also developed based on the MFA and Kernighan-Lin heuristics, respectively.

A significant effort was expended on enforcing hard constraints in the solutions generated by mftp. Mftp can only enforce soft constraints since it uses a penalty function with an associated Lagrange multiplier for each constraint. The {em technique of logical consequences} (tlc) was developed which guarantees the satisfaction of precedence constraints in the scheduling problem without requiring a costly Lagrangean multiplier adjustment phase in the mfa algorithm. Recent investigations into the use of tlc on simpler benchmark optimization problems removed some of the restrictions on the technique. These investigations also showed that tlc can fail to enforce certain non-precedence constraints when the states of the cooperative units in the mftp are highly correlated.

To gain more speed, the optimization methods used in mftp were examined with the intent of reducing their computational complexity. Resulting gains of 20%-30% in computational speed were deemed insufficient when compared to the improvements obtainable by using hierarchy. Increasing the speed of mftp by

using vector and parallel processing was also investigated on Ardent superminicomputers. As with the code optimization, the effort required to increase the speed of mftp by even a factor of five was found excessive. However, increasing the speed of the simpler clustering algorithms was more easily achieved. This may have a multiplicative effect on the speed of the entire scheduler since clustering reduces the problem complexity and greatly increases the speed of convergence of mftp.

Conclusions:

From this experience using the mftp on the scheduling problem, the following conclusion can be made:

1. Mfa can effectively optimize a wide range of forms of cost functions expressed as a function of simple binary decisions;

2. The large number of decision variables slows the convergence of mfa;

3. Applying clustering techniques to hierarchically structure the optimization problem can bring about dramatic increases in speed;

4. Constraints on the problem solutions can be expressed in the cost function using Lagrangean multipliers, but the constraints can be violated as mfa converges. The use of tlc can usually prevent these violations from occurring and reduces the number of Lagrangean multipliers needed and

5. Vector and parallel processing appear effective at increasing the speed of the clustering algorithms used to hierarchically structure the scheduling problem.

Recommendations:

Based upon the previous conclusions, the following areas of research are recommended:

1. The clustering techniques for extracting the hierarchy of the scheduling problem must be improved;

2. Uses of parallelism to speed the clustering must be further investigated and

3. Improvements to tlc must be made to further reduce the possibility of convergence to an infeasible schedule.

Publications:

D. Thomae, D., and D. E. Van den Bout. "Encoding Logical Constraints into Neural Network Cost Functions." To appear in the Proceedings of the IEEE International Conference on Neural Networks, 1990.

Hardware:

None

Software:

The following software modules were developed:

- mftp & the Mean Field Task Planner;
- ganttview and
- an X-11 based graphics program for viewing the schedule output by mftp.

DICE PROGRAM

Task 4.5.4.2 Plasma Spray Disk **GE-Aircraft Engines**

Objective:

The objective of this task was to develop and demonstrate the technical feasibility of using induction plasma deposition (IPD) processing methods to produce advanced lightweight metal matrix composite (MMC) disk subelements for incorporation into gas turbine compressor blisks. The primary challenge faced in this task consisted of establishing a practical manufacturing path for the fabrication of multilayer disk subelements which would produce a composite structure with maximum built-in quality and integrity. Meeting this challenge required the concurrent efforts and input from design, manufacturing, materials processing, behavior and application engineers. Accomplishing this objective with an advanced alpha-2 titanium aluminide (Ti_3Al) matrix reinforced with an SiC fiber would represent a significant state-of-the-art advancement and demonstrate the potential of CE to substantially shorten the development cycle of advanced materials from laboratory to product application.

Approach:

To accomplish the objective of this task the following subtasks were undertaken:

Subtask 1. Subscale MMC Disk Development and Evaluation

In Subtask 1 the titanium aluminide alloy Ti-14Al-21Nb and SCS-6 fiber (a 5.6 mil SiC fiber produced by Textron SMD, Lowell, MA) were used to conduct IPD spray trials in facilities located at GE-CRD, Schenectady NY and GEAE, Lynn MA to develop a wind and spray technique for MMC fabrication. This wind and spray approach is illustrated in Figure 4.5.4.2.1 and consists of plasma spray deposition of the matrix alloy onto a drum followed by machining and grooving to accommodate fiber winding, respraying and repeating the cycle until the desired number of plies have been built up. Each step of the process requires care to avoid contamination during spraying, machining, handling and winding. The plasma spray operation is conducted in an inert (argon) atmosphere which is carefully maintained to minimize interstitial pickup. Figure 4.5.4.2.2 shows the IPD process in operation. The intent of this subtask was to continue the efforts of Phase 1 towards the fabrication of 12" diameter by 32 ply MMC subelements with a fiber volume fraction of 0.30 for evaluation and potential spin test. During this subtask, detailed analysis of each manufacturing step was conducted to determine its impact on MMC cost and quality and to define the experimental work to be performed in Subtask 2 to demonstrate and verify assumptions made about those manufacturing steps.

Subtask 2. Full Scale Disk Development & Evaluation

Subtask 2 efforts concentrated on conducting critical experiments to address manufacturing issues which are anticipated with full scale size hardware (nominally 19-22" diameter MMC's). These experiments were performed to determine if several consolidated MMC hoops could be sized and nested to produce relatively thick multilayer rings which otherwise could not be fabricated by IPD. Trials were also planned to determine the maximum number of plies which could be wind and spray processed before fiber buckling during hot isostatic pressing (HIP) became a problem. Experiments to evaluate HIP tooling methods and their effects on MMC integrity were also performed. A schematic of the planned manufacturing approach which includes HIP consolidation of hoop elements followed by a HIP diffusion bond of several elements is shown in Figure 4.5.4.2.1.

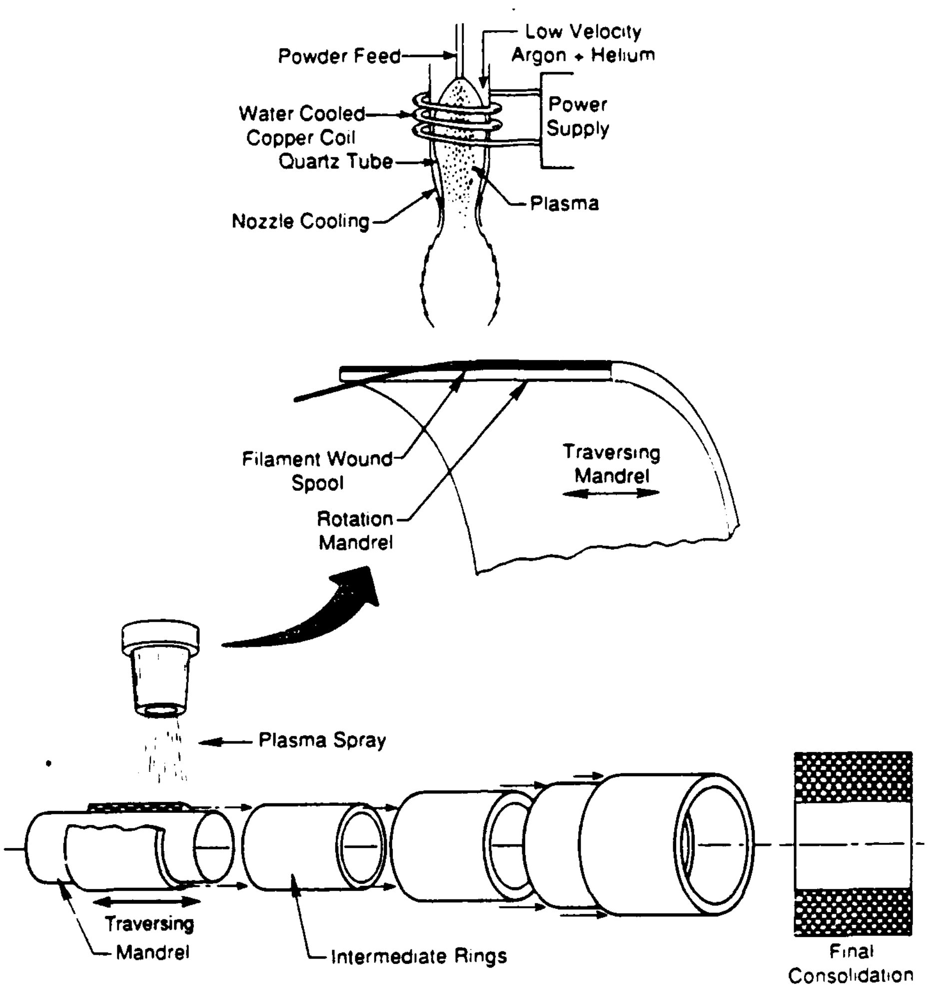


Figure 4.5.4.2.1 Wind and spray approach to MMC subelement fabrication.

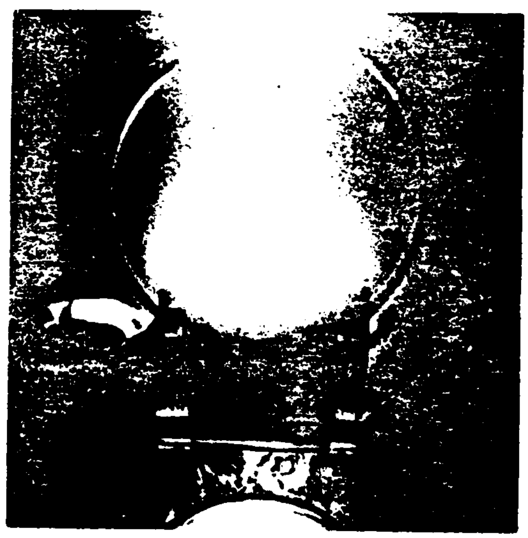


Figure 4.5.4.2.2 Induction plasma deposition (IPD) process in operation.

Technical Results:

Subtask 1. Subscale MMC Disk Development & Evaluation

The initial wind and spray fabrication trials conducted in Phase 1 on 12" diameter resulted in an 8 ply MMC subelement (LPS215) which was HIP'd by Howmet, Whitehall, MI at 1850°F/15 ksi/3 hours. This 5.5" wide consolidated hoop was sectioned and examined for fiber location, alignment and damage. A microfocus x-ray taken through the hoop is shown in Figure 4.5.4.2.3 and illustrtates the excellent fiber alignment achieved. A cross section through this hoop (Figure 4.5.4.2.4) shows the excellent fiber spacing control in the disk axis direction which can be maintained by the wind and spray approach. Improvements are still needed in controlling the radial fiber spacing (ply spacing) to achieve the desired volume fraction. Evaluation of fiber extracted from this hoop revealed no fiber breakage during spray processing or consolidation.

A second, slightly greater than 12" diameter 8 ply hoop (LPS225) was attempeted during this phase to mate with a section of LPS215 to achieve a thicker MMC subcomponent by a nesting approach. The fabrication of this hoop was halted after the fourth ply due to the persistence of delamination during the machining step. In order to better understand the machining characteristics of the Ti-14Al-21Nb alloy, a fundamental evaluation to determine tool loading and wear was conducted by GEAE Manufacturing Center on a 12" diameter plasma sprayed ring. This investigation revealed that a direct correlation exists between tool wear and delamination and that mist cooling along with shallow cuts can reduce the tendency to delaminate.

Subtask 2. Full Scale Disk Development & Evaluation

During Phase 1, several 4" diameter by 4" wide by 4 ply hoops of Ti_3Al/SCS-6 composite were fabricated for demonstrating the feasibility of the nesting approach as it would be needed for full scale MMC subelement manufacture. During Phase 2 these hoops (RF1227 and RF1249) were thermally sized to achieve roundness and then sectioned for nesting trials. Two HIP nesting trials (one with a thin OD can and one with a thick OD can) were conducted at 1832°F/15 ksi/3 hours using the elements shown in Figure 4.5.4.2.5. Both trials were successful in diffusion bonding the inner and outer MMC rings with excellent bond interfaces as shown in Figure 4.5.4.2.6.

Several additional 4" diameter hoops (RF1393 and RF1409) were fabricated using Ti-6Al-2Sn-4Zr-2Mo/SCS-6 as a model system in order to evaluate 1) the maximum potential number of ply buildups before buckling during consolidation and 2) assymetric HIP to control hoop deformation during consolidation. These efforts were incomplete at the end of Phase 2.

Conclusions: (Phases 1 and 2)

1. Wind and spray as an approach for producing Ti_3Al MMC subelements for potential compressor blisks appears to be feasible and offers a unique ability to control fiber spacing.

2. Thermal sizing of MMC hoops which may be distorted from various processing steps is a viable approach to achieving round subelements for possible to produce thicker MMC cylinders.

3. Iterative wind-spray-machine operations to produce cylindrical MMC's can result in separation of the outermost ply when machining. The cause of this problem has been determined to be tool wear and aggressive machining.

4. The wind-spray-machine process does not result in fiber breakage during processing.

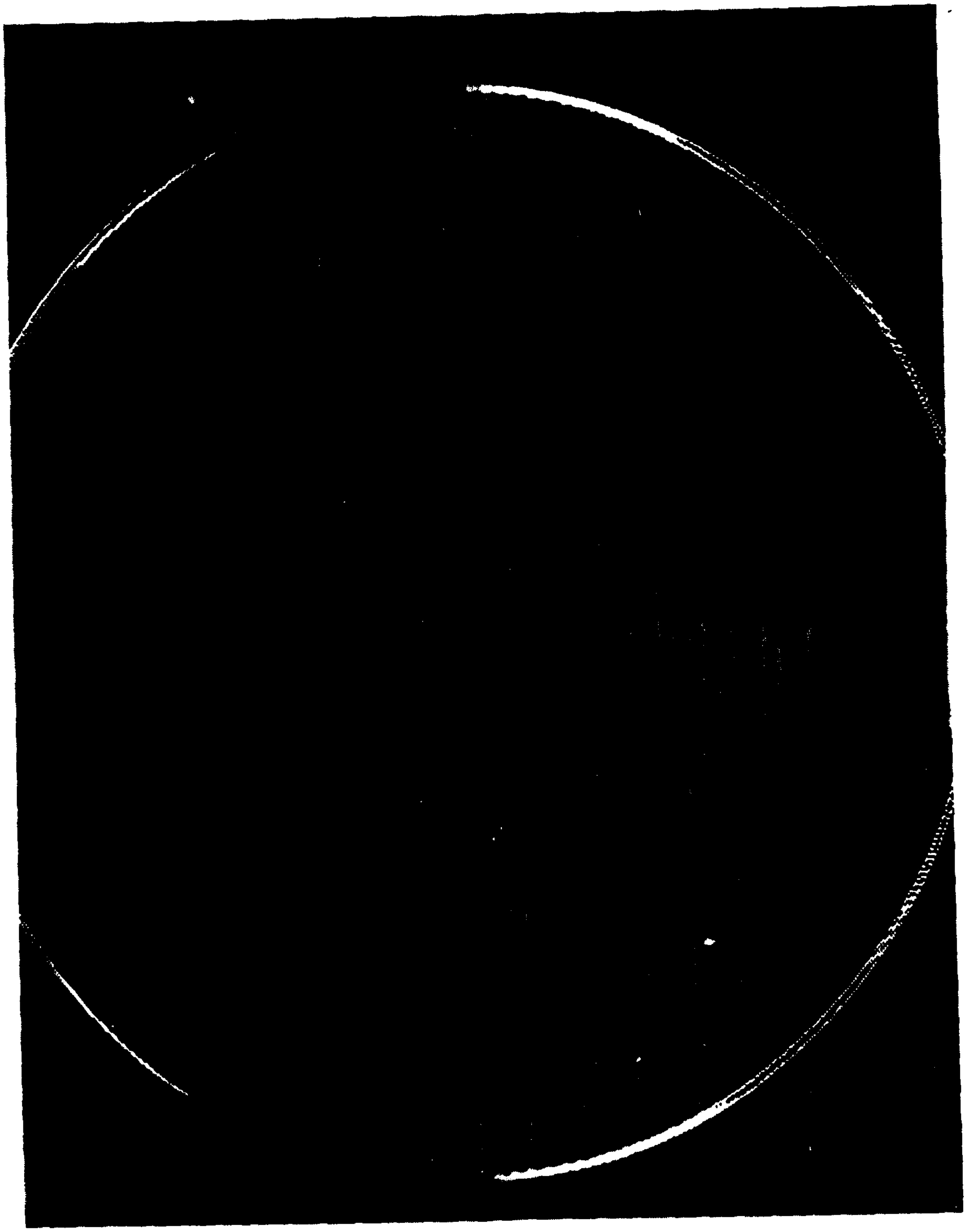


Figure 4.5.4.2.3 Microfocus x-ray of 8 ply MMC hoop LPS215.

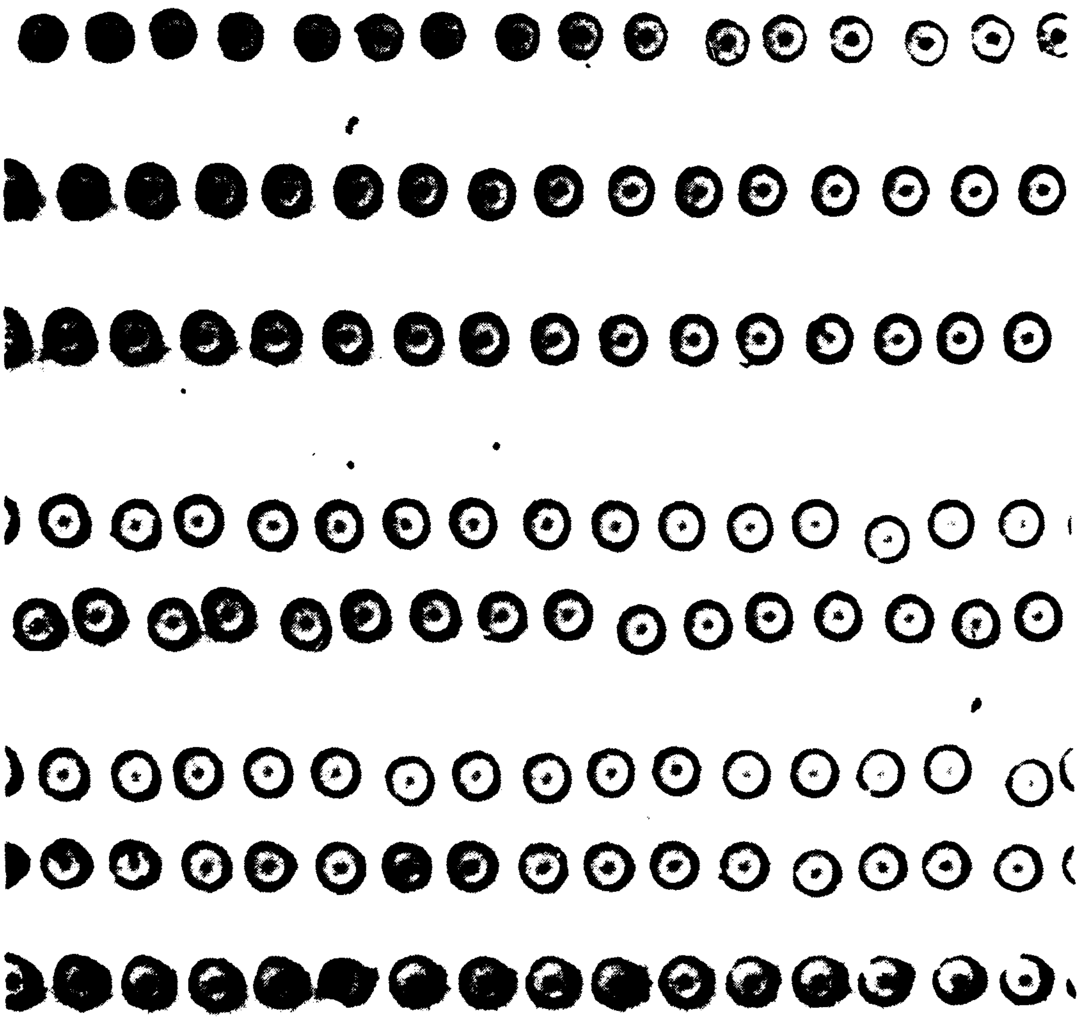


Figure 4.5.4.2.4 Cross section through the 8 ply MMC hoop LPS215.

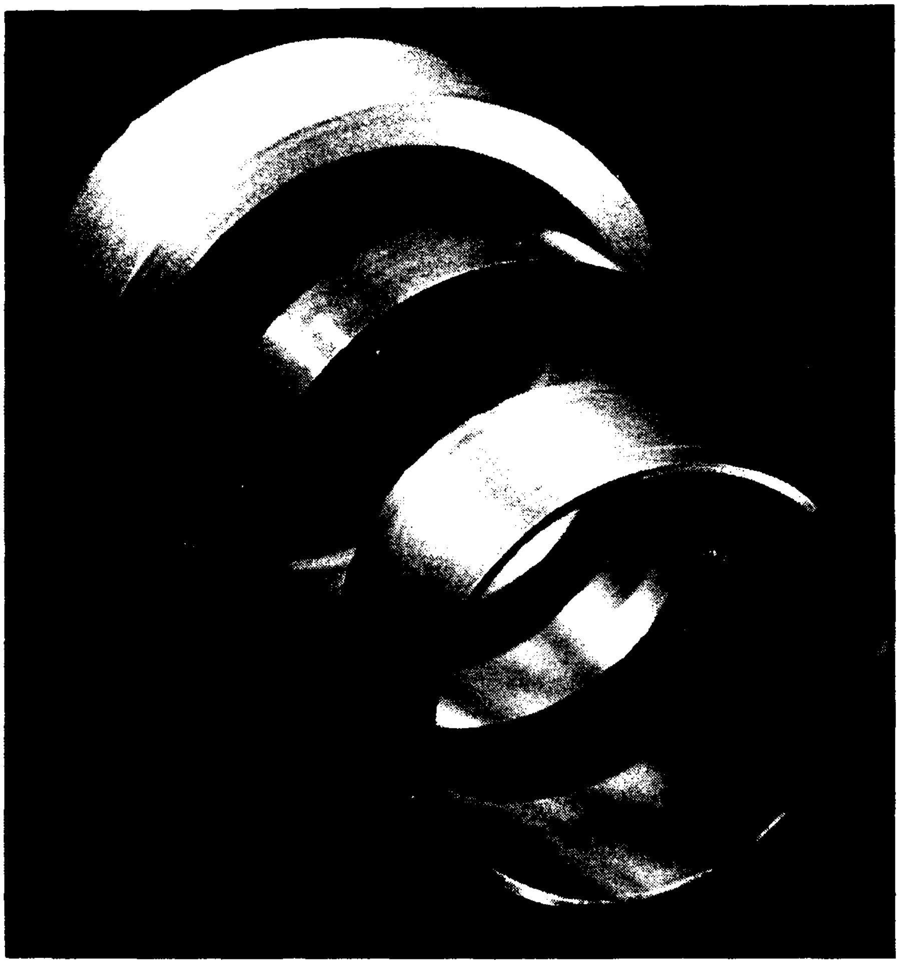


Figure 4.5.4.2.5 Elements of MMC hoop nesting trial prior to assembly and HIP.

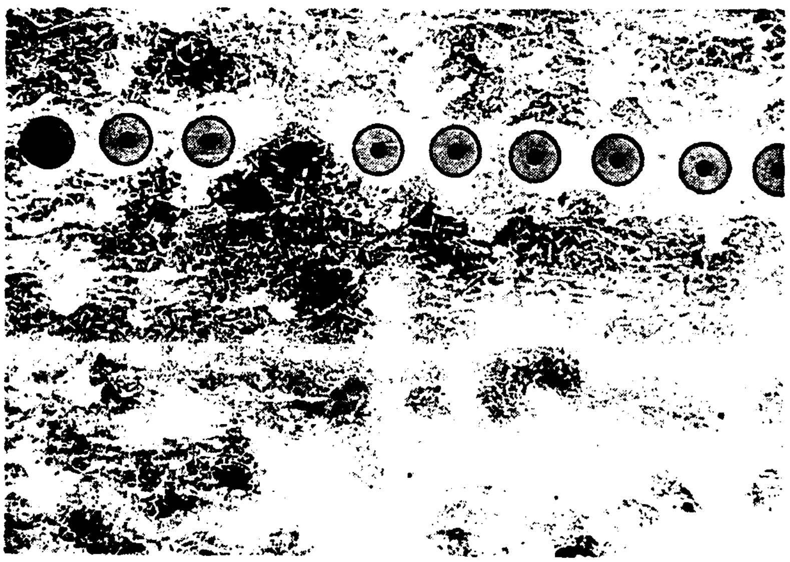


Figure 4.5.4.2.6 Interface between nested MMC hoops after HIP.

5. Nesting of consolidated MMC hoops to produce thicker components appears to be feasible based upon 4" diameter trials.

Recommendations:

Manufacturing methods development should be continued to establish viable techniques for advanced Ti_3Al MMC component fabrication. These techniques should incorporate process control aspects which produce MMC parts with quality built-in to minimize reliance on post fabrication NDE which is extremely difficult in these complex materials. Such process control methods as on-line temperature, mass flow, and environment status monitoring during plasma spray and tool temperature, forces and wear during machining should be developed.

Publications:

There have been no publications resulting from the work performed in this task.

Hardware:

	Item	Purpose	Disposition
1.	LPS180 - 12"D Hoop	Initial large spray trial.	Scrapped.
2.	LPS182 - 12"D Hoop	Second base alloy trial.	Scrapped.
3.	LPS187 - 12"D Hoop	First fiber wound trial, 1 ply, delaminated.	Cut up for evaluation.
4.	LPS197 - 12"D Hoop	Base alloy hoop to replace LPS187, too thin.	Scrapped.
5.	LPS215 - 12"D Hoop	8 ply fiber wound hoop.	Evaluated in Phase 2.
6.	LPS225 - 12.5"D Hoop	4 ply MMC hoop.	Evaluated in Phase 2.
7.	LPS279 - 12.7"D Hoop	Base alloy hoop of Ti-6-4.	Used as base LPS287.
8.	LPS287 - 13.2"D Hoop	Ti3Al build-up on LPS279 for intermediate hoop.	Used for machining studies in Phase 2.
9.	LPS288 - 13.2"D Hoop	2 ply Ti3Al MMC hoop for outer hoop.	Fabrication discontinued due to Phase 3 changes.
10.	LPS292 - 4"W Monotape	Winding calibration trial using Ti-6-4/SCS-6.	Cut up for evaluation.
11.	RF1122 - 4"D Hoop	1 ply initial trial hoop with 50 mil base, cracked during machining.	Used for sizing trials.

12.	RF1144 - 4"D Hoop	1 ply sprayed on solid steel mandrel, unsuccessful due to CTE mismatch problems, debonded during cooldown.	Scrapped.
13.	RF1227 - 4"D Hoop	4 ply fiber wound hoop on 0.125" base.	Used for nesting trials in Phase 2.
14.	RF1243 - 4"D Hoop	2 ply fiber wound trial due, unsuccessful due to high interstitial pickup.	Scrapped.
15.	RF1249 - 4"D Hoop	4 ply fiber wound hoop on 0.125" base.	Used for nesting trials in Phase 2.
16.	RF1393 - 4"D Hoop	8 ply Ti-6242/SCS-6 fiber wound hoop for buckling studies.	Sectioned for evalulation.
17.	RF1409 - 4"D Hoop	4 ply Ti-6242/SCS-6 fiber wound hoop for assymetric HIP trials.	Sectioned and awaiting HIP.

DICE PROGRAM

Task 4.5.6 Assembly Demonstration — Cimflex

The objective of the ongoing DICE/HDE effort is to adapt the DICE architecture to support engineering teams involved in the process of designing, manufacturing and testing high-density electronics (HDE) products, specifically electronic component assemblies (ECAs). The DICE methodology is expected to improve dramatically both the efficiency and effectiveness of this process. The efficiency can be measured in terms of the level of resources required and the productivity of individual engineers. The effectiveness can be measured in terms of the speed with which the process is executed, and the quality of the ECAs produced.

The central methodology utilized in the DICE program is that of "concurrent engineering," which increases the degree of parallelism and coordination among the various tasks involved in ECA production (design, layout, assembly, test, etc.). This approach has a number of important beneficial impacts, in that it enables the engineering team to:

- comprehend and articulate better the many relationships and trade-offs among product requirements and alternate technical approaches
- detect inconsistent requirements or design problems earlier in the process, and apply collective knowledge to resolve them
- schedule tasks in accordance with their dependencies on the results of other tasks rather than in a serial manner.

The DICE electronics application incorporates a number of methods that were either supplied by the DICE architecture or developed specifically for the HDE domain. These methods, and the tools and services that support them, include:

- Concurrency management and communication services enabled by the shared PPO data base and the Local Concurrency Manager (from GE/WVU).
- Requirements tracking and fulfillment, enabled by the Requirements Manager system.
- Early evaluation of test strategy, and generation and assessment of test plans, enabled by the Design for Testability advisor.
- Integration of component engineering data, enabled by the Component Manager subsystem and the Component Engineering Assistant.
- Feedback of manufacturability advice to design engineers, enabled by the workcell IPP interface (from WVU) and the Component Engineering Assistant.
- Flexible, rapid prototype assembly enabled by the workcell IPP interface and the Assembly Planner and Simulator (from WVU).

To put these methods into practice, the DICE tools were integrated with a suite of commercial CAE/CAD software (from Valid Logic), including a schematic editor and a layout package. The methods and data flow are summarized in Figure 1, and the tools and architecture are depicted in Figure 2. Section 2 below describes the current state of electronics-related CAD/CAE technology, and the DICE program opportunities. Section 3 presents the scenario that was used to demonstrate the HDE application on December 15, 1989. Additional details about the tools that were developed are provided in Section 4. Finally, Section 5 summarizes the accomplishments of the program.

Figure 1: Electronic Methods Data Flow Diagram

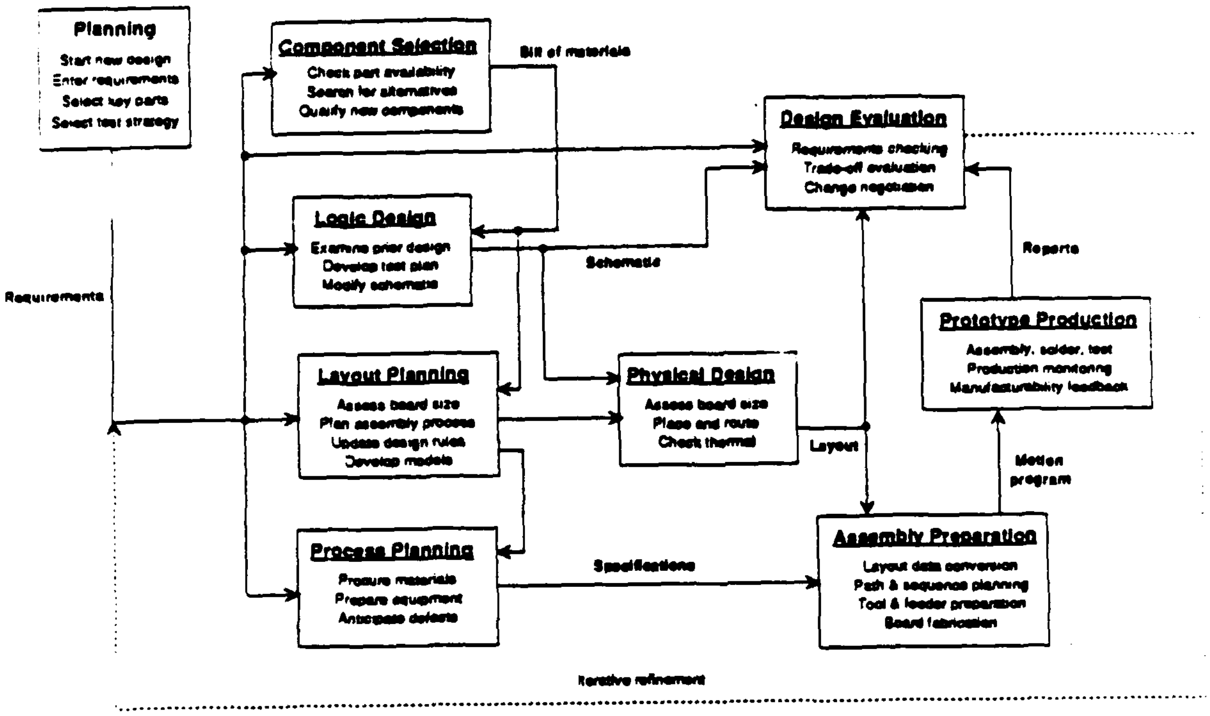
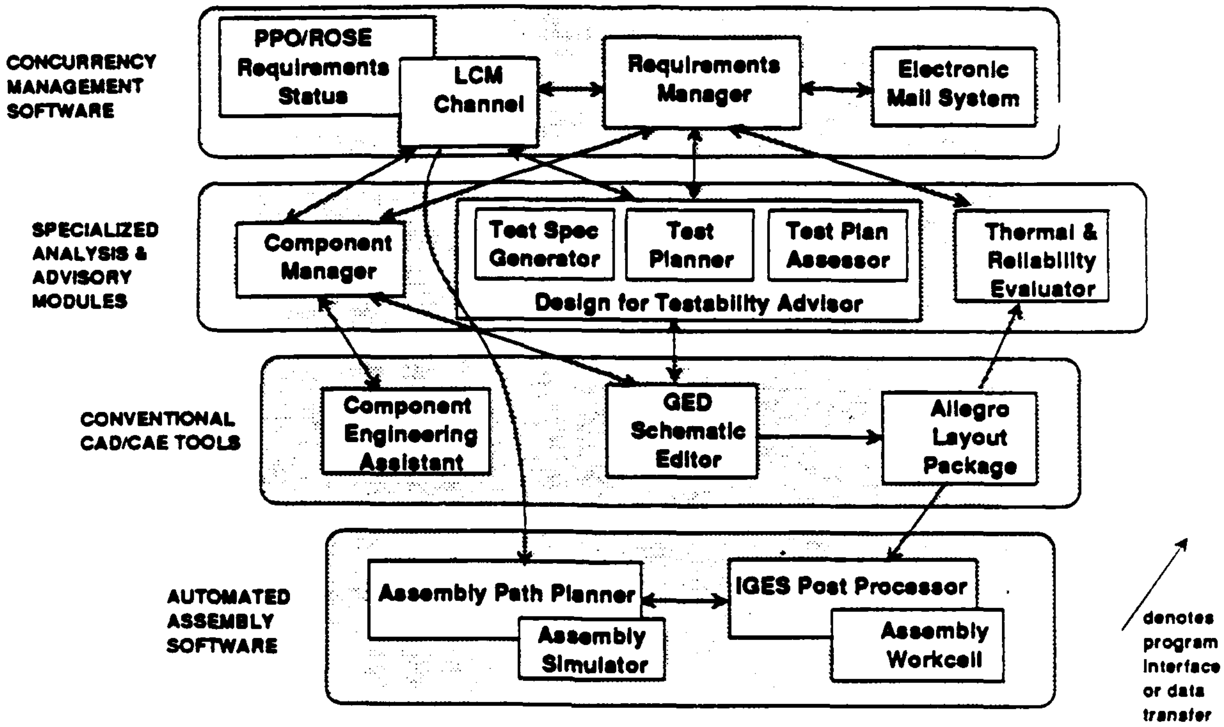


Figure 2: HDE Architecture Diagram



2. Potential Contribution of DICE to CAD/CAE

The field of electronic engineering is relatively advanced in terms of the availability of software tools to support the development of new products, ranging from application-specific integrated circuits to complex instruments. The initial application and demonstration area selected for the DICE program is high-density electronic component assemblies (ECAs); these are primarily multi-layer, surface-mount circuit boards with about 60 to 160 leads per component, typically used in the computer and aerospace industries. ECAs are generally developed through the following sequence of steps:

Functional Specification

Usually derived from overall system requirements; represented as a "black-box" model.

Circuit Logic Design

Involves selection and connection of components, and development of schematic representation.

Circuit Analysis

Validates functional behavior, timing, etc. using detailed simulation techniques; verifies conformance with electrical design rules.

Board Design and Layout

Physical specs (e.g. packaging; connectors; use) are derived from system requirements; this step involves detailed component specification, gate assignment, placement and routing, governed by design rules to ensure *manufacturability*.

Board Analysis

Review of the design for testability, reliability, and manufacturability, sometimes resulting in the need for modification of the circuit or the layout.

Board Fabrication and Assembly

Based on drill tape, router tape, and art work developed in the board design and layout.

In small organizations, many of these steps are frequently blurred together, as there is considerable dialogue within engineering teams regarding the downstream implications of design choices. However, in large organizations, a tendency has emerged for these steps to be performed sequentially by separate teams, often with limited communication between the teams. As a consequence, a printed circuit board may go through several lengthy iterations of the engineering cycle before it is deemed satisfactory. Thus, the DICE approach to concurrent engineering has the potential to both reduce the cycle duration while improving product quality.

There currently exist a number of commercial software aids corresponding to each of the above steps. These tools generally run on powerful workstations and personal computers, and consist of a suite of programs that enable engineers to design, analyze, and modify the board interactively. In conventional terminology, the circuit design and analysis stage is referred to as computer-aided engineering (CAE), the board design and analysis stage is referred to as computer-aided design (CAD), and the production stage is referred to as computer-aided manufacturing (CAM). The present generation of tools closely matches the sequential model of the engineering cycle; that is, the output of one tool is used as input to the next.

In recent years, the CAE/CAD/CAM industry has moved toward adoption of hardware and software standards that enable customers to utilize a mix of tools from different vendors. These standards include the UNIX and DOS operating systems, Ethernet, TCP/IP, and NFS network protocols, and EDIF and IGES data exchange formats. Vendors have also sought to enhance ease of use by introducing graphics-based interfaces with selectable icons and multi-window displays. Local area networks are now used extensively to permit sharing of information and efficient allocation of computing resources.

Given this context, and based on discussions with engineers as well as a review of the literature, we gained an understanding from a user perspective of desirable system functionality that is not currently available. This has led us to focus on a number of areas in which we believe the DICE program can make a contribution to ECA engineering and design:

- Tracking of functional requirements through the engineering cycle, evaluation of the extent to which designs will meet these requirements, and analysis of the associated trade-offs
- Maintenance of functional information about various blocks of a circuit schematic, and use of this information to guide circuit modification and subsequent design choices
- Availability of advisory information that can assist a circuit or board designer in an interactive, incremental fashion, based upon downstream constraints and specialized knowledge
- Capability for engineers to capture, edit, and maintain their specialized knowledge, for purposes of providing automated advice to others in the appropriate context
- Maintenance of consistency within a complex design as specific portions are modified, possibly by different engineers, and propagation of change information throughout the team
- Configuration management to support team coordination in the partitioning, refinement, and merging of design specifications, including alternative approaches
- Capture of design history and rationale in a form that can be interpreted automatically to identify problems or opportunities that were encountered previously
- Translation of design specifications into process specifications for purposes of rapid, reliable prototype assembly
- Analysis of product quality and performance in relation to product engineering, for purposes of feedback to designers
- Our view is that the primary value of DICE is not in introducing new tools to support specific functions, but rather to provide a unifying infrastructure in which existing tools can be used more effectively. The demonstration scenario presented below was developed to illustrate how the DICE/HDE application addresses these identified needs.

3. Demonstration Scenario

3.1. Introduction

The electronics demonstration showed an application of the DICE architecture to a hypothetical project, in which an existing DOD electronic component assembly (ECA) is modified in order to achieve several objectives:

- deliver the same function with half the board size
- improve testability using automated test equipment
- complete first production within within 30 days.

The project is assumed to take place in a large engineering organization. The project team reduces the

board size using surface-mount technology, and exploits the DICE concurrent engineering environment to move from concept to product as rapidly as possible, like a "virtual tiger team."

While the project is hypothetical, the board that was selected is a real one obtained from the Navy. It is called an Embeddable Standard Avionics Processor (ESAP), and the Navy was interested in reducing its size from SEM-E, i.e. 6" X 6" down to SEM-C, i.e. 3.5" x 6". The steps portrayed in the demonstration reflect authentic design and engineering activities that Cimflex Teknowledge undertook to create a new ESAP_C board using surface-mount devices.

The demonstration consists of about twenty scenes representing the activities of various project team members during the 30-day duration of the project. It shows how the DICE tools and architecture enable them to coordinate the fulfillment of product requirements, to recognize potential problems, and to communicate effectively as they perform mid-course corrections in order to meet a tight deadline. The concurrent engineering process implies that the project team will address simultaneously a number of different concerns, including:

- Functionality
- Testability
- Reliability
- Manufacturability
- Maintainability
- Affordability

There are five major functional roles portrayed in this scenario, each using a different workstation:

Chief Engineer	Sun 3/60
Logic Designer	Sun 3/60
Layout Engineer	Sun 4/110
Component Engineer	IBM PS-2
Manufacturing Engineer	Silicon Graphics & IBM PS-2

Several parallel threads of activity converge during the demonstration. As in any real-life engineering organization, there are many activities being carried on concurrently. Background work, such as qualification of new components and processes, is continually being done in anticipation of future requirements. For time-critical projects, it is essential that the team be able to draw upon these existing resources, and the DICE environment provides an information management framework in which these different threads of activity can be effectively woven together.

The project team starts with the existing ESAP design and redesigns it to fit in half the area, using components in fine-pitch SMT packages and making other changes as needed. The choice of components is guided by cost, availability and size considerations based on up-to-the-minute information provided by DICE services. Testability is improved in the new design, in spite of the fact that in-circuit test (probing) is not feasible on the smaller-size board. Finally, after the design is completed and the board placed and routed, the layout data are translated into assembly instructions for an automated workcell.

3.2. Requirements Development Phase

The Chief Engineer launches the project by invoking the Requirements Manager (RM) and assigning the new design a name (ESAP_C). RM allows the Chief Engineer to incorporate standard sets of company requirements (e.g. MIL-SPEC) and to specify additional requirements, such as:

fault coverage:	must be 95%
package type:	must be SMT

component availability: must be "today"

The list of requirements that the Chief Engineer is creating will be used throughout the project to check the status of the board design automatically as it evolves. If necessary, these requirements can be modified easily by authorized personnel. The requirements are stored in the PPO so that they can be accessed by any of the DICE tools.

Immediately upon notification from the Chief Engineer, the Logic Designer assigned to the project pulls up the old schematic and begins to examine the requirements. The Logic Designer notes the size reduction objective, and then focuses on the testability requirement, thinking that this will be a challenging area.

Meanwhile, the Layout Engineer has been working on refinement of a new process called "bottom-side assembly," in which passive SMT components are placed on the bottom side of the board with adhesive and then wave-soldered. The Layout Engineer has received some negative feedback from manufacturing, based on a prototype run in which a number of components fell off the board, so he is modifying the package symbol (i.e. land pattern) to increase the adhesive dot size. Although the Layout Engineer presently is unaware of the ESAP_C project, this new bottom-side assembly process ultimately will prove vital in meeting the project objectives.

The Component Engineer is involved in qualifying for another project a new component, a 1755 chip with 16-bit CPU and buffers. The Component Engineering Assistant (CEA) tool reviews the component qualification schedule, showing when each department will complete its work. The 1755 ultimately plays a crucial role in the ESAP_C project, but at this point it is not even under consideration. The Component Engineer, because of pressure from another project, is trying to accelerate the qualification process on this component. Finding that purchasing is on the critical path, the Component Engineer decides to contact the Purchasing department to see if they can expedite procurement of this component. Purchasing uses the same CEA tool to review their schedule commitments.

3.3. Testability Evaluation Phase

The Chief Engineer invokes the Test Specification Generator (TSG) module of the Design for Testability (DFT) advisor, which obtains the current set of requirements from the PPO. TSG recommends a board-edge test strategy based on the board size, SMT packaging assumptions, and ATE equipment specified in the requirements. The Chief Engineer accepts the suggested strategy, and causes a new requirement to be added to the PPO. Thus, TSG has applied specialized knowledge about test engineering to influence an early design decision.

At this point the Layout Engineer works on creating a new package description for a 100-pin SMD to support the 1755 component. Specifying packages is just one of the many component qualification activities that must be completed before a new component is approved for incorporation into a design.

The Logic Designer continues working on testability; focusing on the memory block, the Logic Designer uses the Test Planner module of the DFT advisor to generate a test plan for that block. The Test Planner takes into account the board-edge test specification that was previously accepted by the Chief Engineer.

3.4. Initial Design Phase

The Chief Engineer uses the Component Manager (CM) to substitute key SMT components into the proposed bill of materials (BOM). Starting with the BOM for the old design, the Chief Engineer selects the 1750 component and indicates a desire to search for any components having the same function (i.e. part number) and matching the product requirements stored in the PPO. CM then searches the component qualification data base, which contains up-to-date information maintained by component engineers. CM displays a list of components that match the target characteristics. The Chief Engineer then selects the desired component, and CM automatically substitutes it into the BOM. The Chief Engineer also annotates this design choice with an explicit rationale.

The Chief Engineer now searches for a recently announced component, the 1757. Its absence from the corporate database prevents specifying an unauthorized part that would later cause problems. After specifying a few other key components, the Chief Engineer passes the BOM on to the Logic Designer. The Logic Designer now continues development of the BOM by finding SMT components equivalent to those used formerly. The Logic Designer goes through the same procedure for substituting a component into the BOM, and at the same time replaces the original component in the schematic by its substitute. When finished, the Logic Designer passes the proposed BOM to the Layout Engineer, asking for a preliminary feasibility assessment.

The Layout Engineer does a trial placement of the board using the existing BOM and notes that it will probably not fit on a SEM-C. The Layout Engineer sends back a message recommending the use a two-sided board, placing some components on the bottom side. The Layout Engineer notes that the recent evaluations of bottom-side assembly indicates it offers significant advantages.

Meanwhile, unrelated to the ESAP_C project, a manufacturing engineer who is responsible for SMT assembly is working on the 1755. This engineer enters a note regarding lead coplanarity problems recently observed with the 1755 and the expected defect rate. The problem shows up during assembly as a vision-based rejection. The Manufacturing Engineer has not yet begun work on the ESAP_C board, but has grown accustomed to capturing experience in the corporate memory for the benefit of future projects.

3.5. Design Evaluation & Refinement Phase

The Logic Designer has produced a first-cut design, and is now checking the various functional blocks for testability. He is working on the memory block, and runs the Test Plan Assessment module of the DFT advisor in order to estimate fault coverage. The Logic Designer, having little practical test engineering experience, appreciates getting these early estimates of fault coverage. The DFT reports that there will be observability problems and suggests a change to the design that could improve fault coverage: adding an edge connector for a particular pin. The Logic Designer adopts the recommended change.

Having achieved what appears to be the first satisfactory logic design, the Logic Designer now releases Rev 1.0 of the design to the Layout Engineer and also requests that Manufacturing do a preliminary check because of the novelty of this design. The Layout Engineer uses autoplacement to lay out the board, a process that takes several hours. In the demonstration, the layout results indicate the board almost fits on a SEM-C, but not quite. The Layout Engineer sends a warning message to the rest of the

team, and also transfers the layout data to the Manufacturing Engineer for evaluation. The Manufacturing Engineer uses the IPP workcell interface tool to transform the layout data, which is in IGES format, into a set of placement instructions for the automated workcell. IPP notes that a fiducial is missing, which the Manufacturing Engineer reports to Layout.

The Chief Engineer learns about the board size problem and starts to get worried. Swinging into crisis mode, the Chief Engineer searches for an alternate component that might offer a solution. Knowing that the 1750A and its buffers are consuming a good deal of real estate, the Chief Engineer uses CM to search for any component in the 1750 class regardless of other constraints. This identifies three candidates that had not been considered previously because of their inability to satisfy the availability constraint. The 1755 component appears particularly attractive. Looking at the detailed schedule, the Chief Engineer discovers that the 1755 has almost completed the qualification process and sends urgent mail to procurement asking them to expedite the process. He also launches an alternate design effort using the new 1755 component.

3.6. Design Completion and Assembly Phase

After another week, things begin to fall into place. The first design did not fit on the board, but with the 1755 the Logic Designer was able to consolidate several devices into one package and reduce the required board area. At the same time, the 1755 has received higher priority, and is now expected to be qualified in time. The design is now ready for assembly preparation.

The Layout Engineer places and routes the Rev 2.0 board and finds that it does fit on a SEM-C board. A thermal check is run to see whether the reliability requirement is satisfied, and the board passes. The physical design data is then output from Valid Allegro in Autocad DXF format. Autocad's utility, running on a PC, translates DXF into IGES. The Manufacturing Engineer uses the Workcell Interface IPP to translate IGES into Adept's ADX format. The ADX file is sent to a Silicon Graphics workstation, where the Assembly Planner module optimizes the placement sequence. Based on the placement sequence and knowledge of the workcell capabilities, the Manufacturing Engineer now has APS synthesize a motion program for the assembly. The Assembly Simulator displays the motions on the screen. The Workcell Interface transmits the program to the workcell.

The Chief Engineer runs the Requirements Manager on the final design to ensure that the product actually meets all the requirements. Indeed, it passes. The Logic designer releases Rev 2.0 and has the documentation package produced electronically. Finally the Manufacturing Engineer downloads the assembly plan to the flexible assembly workcell, which is driven by CAD output. The Manufacturing Engineer will monitor the assembly process, and may recommend further design changes to deal with quality problems. Thanks to the automatic data conversion, such changes can be implemented readily.

4. Electronics Application Tools

4.1. Introduction

The following tools were developed by the Knowledge Systems Division of Cimflex Teknowledge as part of the HDE application project. Figure 2 depicts the relationship between these tools and those that were developed by other groups and integrated into the application.

4.2. Requirements Manager

The Requirements Manager (RM) system is a key component of the DICE/HDE application. This system provides engineers with the capability to capture and manage the product requirements, specifications and constraints which influence their technical decisions. It makes the trade-offs and issues involved in satisfying requirements more understandable and controllable, by providing a shared electronic medium for explicit representation, maintenance and validation of constraints.

The RM human interface is implemented in Sunview. All requirements information is stored in the PPO, and can be shared with other DICE tools. Requirements evaluation may trigger a call to a DICE tool that performs the evaluation (e.g., DFT advisor).

The RM system provides a facility for recording and testing constraints and assumptions, supporting their validation, and resolving difficulties encountered in meeting the requirements. Specifically, the RM provides the following types of capabilities:

- Capturing and tracking functional requirements
- Formulating design, manufacturing and test specifications
- Coordinating team members relative to design specifications
- Checking compliance with requirements and specifications
- Supporting negotiation regarding design change trade-offs
- Propagating effects of changes in requirements and specifications

The RM has two basic types of functionality: requirements browsing and editing, and requirements evaluation. Requirements are organized into categories, such as testability, reliability, etc. The user can examine all categories in a summary table, and then can work with a specific category in greater detail.

Appendix A provides a functional overview of RM Version 2.0, to be implemented in 1990.

4.3. Component Manager

The Component Manager (CM) was developed as a demonstration module in order to highlight a commercial requirement which we expect will be filled in the next few years. It provides a link between the design engineering side and the component engineering side of an organization, by enabling design engineers to specify the desired characteristics of electronic components and search for matches in a data base of approved components. Thus, it facilitates the development of a bill of materials that is consistent with the published product requirements.

The CM human interface is implemented in Sunview. It is capable of reading requirements information stored in the PPO and component information stored in a component qualification data base. It can output selected components to a bill of materials (BOM) file.

CM can search for components based on target characteristics derived either the product requirements stored in the PPO or from an existing BOM. The user can modify or augment this target pattern. CM then searches the component qualification data base, which contains up-to-date information maintained by component engineers. In this way a design team can screen components before investing a great deal of time in a design that might not be feasible.

Upon completing a search, CM displays a list of components that match the target characteristics.

Once a component is selected, the user can instruct CM to insert or substitute components into the proposed BOM. The user can also annotate the entry with an explanation or rationale.

4.4. Valid Logic Tools

We obtained a variety of commercial ECAD/ECAE tools from Valid Logic, Inc. and transferred these to the CERC at no cost to the DICE program, as described below.

Schematic Editor

The GED schematic editor captures graphical, hierarchical representations of logic design schematics, and permits them to be edited visually. GED reads and writes to a number of files in proprietary formats, including a logic symbol file, a hierarchical logic design file, and a flattened logic design file. GED provides a range of visual browsing and editing functions. It can search for individual components, and zoom in and out to display portions of a design at different scales. The user modifies a design by selecting components from the library, placing them on the schematic, and connecting them to existing components or connectors.

Layout CAD Tools

Allegro is an integrated electronic CAD package that supports all phases of layout engineering. It is used to create a physical layout from a flattened schematic, and provides precise visual displays of the component placement and routing. The main data base utilized by Allegro is the Physical Product Database, which contains detailed descriptions of the layout for each design. Allegro also accesses a physical component data base and a design-rule database. Allegro supports a variety of manufacturing processes, including multi-layered through-hole and surface-mount technologies. It combines a number of functions that are closely integrated:

- Design rule editing and checking
- Package symbol characterization
- Automatic or semi-automatic placement and routing
- Thermal and reliability evaluation
- Mask data generation
- Query and report generation

4.5. Component Engineering Assistant

Like the Component Manager, The Component Engineering Assistant (CEA) was developed as a demonstration module to suggest how component engineers and others might review and manage the qualification schedules for various components. CEA is implemented in LIBRA, which is a knowledge maintenance facility built on top of M.1, Cimflex Teknowledge's PC-based expert system shell.

CEA is used to view the qualification schedule for selected components, which provides information about the various parties involved and their completion dates. These dates can be edited by the user, and CEA calculates the earliest overall completion date. CEA can also display the set of components for which a particular engineer has outstanding commitments.

5. Summary of Deliverables

The objective of CTC's FY89 work was to develop a preliminary version of a HDE workstation environment that enables electronic engineers to utilize conventional CAE/CAD tools and also benefit from the concurrency and information management facilities of DICE. The HDE environment is designed to support teams of engineers responsible for the multi-stage process of ECA development, enabling the capture and re-application of high-level knowledge about constraints and design alternatives related to cost, performance, testability, manufacturability, and other important considerations. Our FY89 demonstration effort focused upon three types of support capabilities:

- requirements tracking and change notification,
- advice about testability for a design schematic, and
- coupling with an automated manufacturing workcell.

The specific software deliverable for FY89 was an initial functional design and implementation for a workstation environment that integrates DICE architectural services with knowledge-based advisors and conventional CAD/CAE tools. The environment includes:

- a set of commercial CAE/CAD tools from Valid Logic, Inc., including commonly used tools such as a schematic editor, fault and timing simulation, and physical layout package
- DICE architectural services that were tested and integrated (PPO/ROSE and LCM)
- the Requirements Manager system
- the knowledge-based Design for Testability advisor
- assembly planning and simulation tools, as well as a workcell data preparation facility, developed by WVU.

In addition, we delivered summary documentation for these tools and a demonstration guide, which was incorporated into the DICE demonstration package produced by GE.

References

1. J.K. Hollomon, Jr., *Surface-Mount Technology, for PC Board Design,* H.W. Sams, Indianapolis, 1989.
2. P. Horowitz & W. Hill, *The Art of Electronics,* Press Syndicate of the University of Cambridge, Cambridge, 1989.
3. K.P. Parker, *Integrating Design and Test: Using CAE Tools for ATE Programming,* Computer Society Press of the IEEE, Washington, D.C., 1987.
4. R.P. Prasad, *Surface Mount Technology, Principles and Practice,* Van Nostrand Reinhold, New York, 1989.
5. V. Solberg, *Design Guidelines for Surface Mount Technology,* Tab Books, Blue Ridge Summit, PA, 1990.
6. R. Spence & R.S. Soin, *Tolerance Design of Electronic Circuits,* Addison-Wesley, Reading, MA, 1988.

Appendix A

REQUIREMENTS MANAGER FUNCTIONAL OVERVIEW

1. Background

Requirements play a key role in shaping and guiding any product development effort. The role of the Requirements Manager (RM) within the DICE architecture is to provide the engineering team with the capability to capture and manage the product requirements, specifications, policies, rules, and constraints that influence their technical decisions. It clarifies product development trade-offs and issues by providing a shared electronic medium for the representation, maintenance and validation of requirements. This document provides a functional specification for RM Version 2.0, to be implemented in 1990.

The examples used in this document are drawn from the initial application area for the Requirements Manager, namely high-density electronic component assembly (ECA) products; these are primarily multi-layer, surface-mount boards with roughly 60 to 160 leads per component, typically used in the computer and aerospace industries. ECA's are generally developed through a cycle of steps that include circuit design, analysis, layout production, and testing. It should be noted, however, that the RM system provides a generic capability that can be extended readily to other product domains and activity cycles.

2. Purpose of the Requirements Manager

The RM is a tool intended to support concurrent engineering by providing a project leader and his/her team with the capability to capture, manage, and verify product requirements. Figure A-1 depicts the manner in which RM interacts with other tools and with various users on a project team. There are four major type of capabilities illustrated:

1. Requirements are defined explicitly and disseminated to the project team. Requirements are expressed in a user-definable hierarchy, maintained through a graphical human interface, and are stored in a shared knowledge base with appropriate protection mechanisms. Thus they may be viewed selectively by various specialists, and modified directly by authorized personnel.

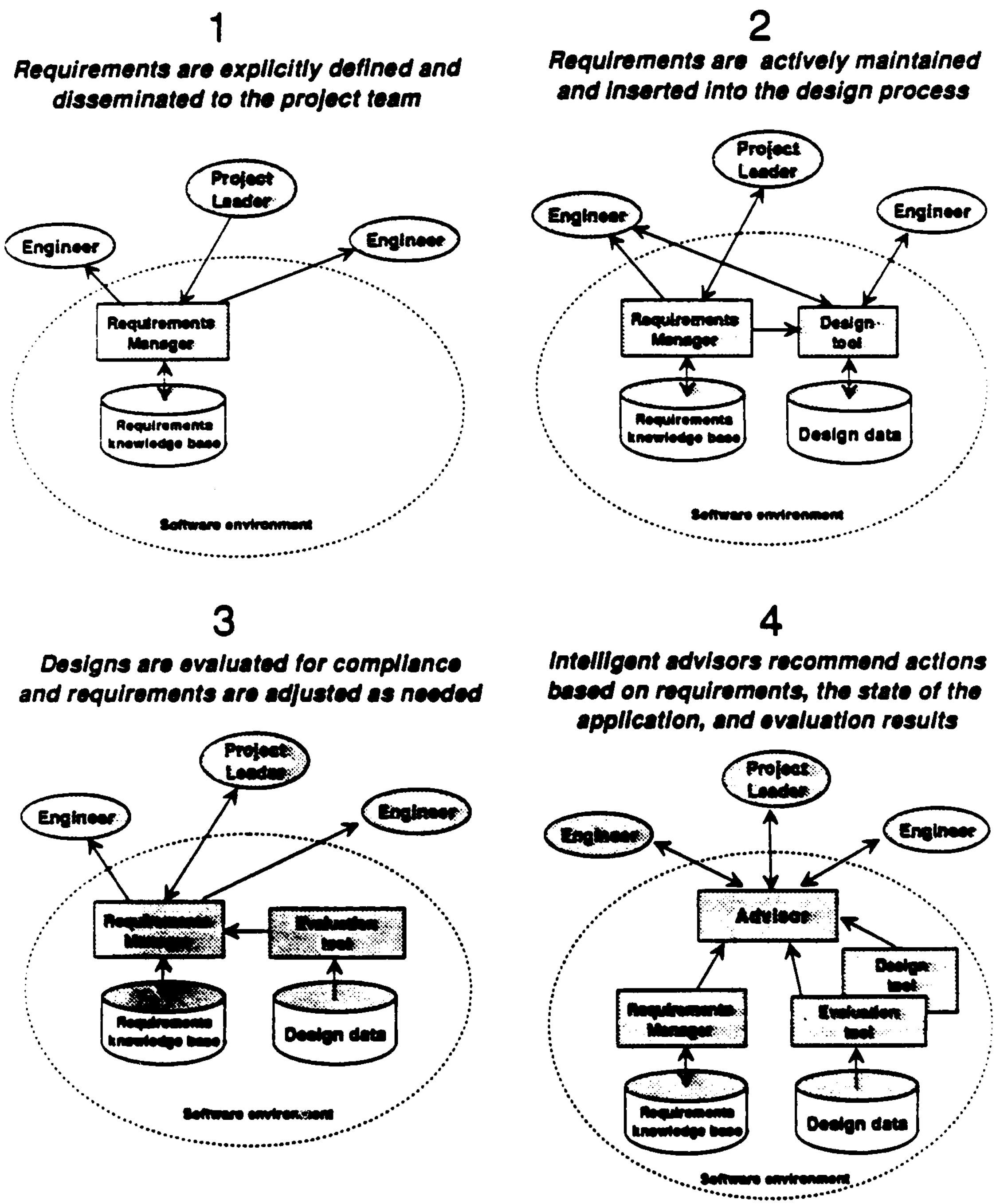
2. Requirements are applied actively and maintained during the design process. The project leader or other authorized personnel may change or re-structure the official requirements as the project proceeds. Various design tools (e.g. CAD layout tools) can extract the current set of requirements to guide their own operation when appropriate.

3. As design activities progress, the RM can be invoked to check compliance, and it can in turn invoke evaluator tools and interpret their results. RM will provide useful information about requirements including pass/fail status, history information, ownership, time-last checked, documentation, and results from evaluation or advice. Persistent storage of evaluation input conditions and results encourage use of parametric requirement expressions.

4. RM can interact with intelligent advisors, which recommend actions to improve a product design based upon the existing requirements, the state of the process, and the most recent evaluation results.

RM will be used iteratively to track the status of a project as the evolving design is compared with the customer requirements and repeatedly refined. The RM system will provide a facility for recording and testing requirements and specifications, supporting their validation, and resolving difficulties encountered in meeting the requirements. Specifically, the RM seeks to provide a product development team with the following capabilities:

Figure A-1: Using the Requirements Manager



-Capturing and tracking product requirements
-Coordinating team members in *refining requirements*
-Propagating effects of changes in requirements
-Checking compliance with requirements
-Facilitating negotiation regarding design tradeoffs

The anticipated benefits of integrating the RM capability into the DICE/HDE environment fall into two broad categories:

First, we expect that RM will facilitate concurrent engineering in the overall ECA design and manufacturing process, through:

- providing a precise language and medium for articulating, maintaining, tracking and enforcing requirements
- providing a group mechanism for identifying design trade-offs and supporting change decisions
- assuring compliance with corporate policies
- accumulating a history of experience in the fulfillment of requirements for various products.

Second, we expect that RM will improve individual performance within the stages of the ECA design and manufacturing process, through:

- assuring consistent communication of relevant requirements for each stage of the process
- facilitating the validation of partially-completed designs relative to requirements.

These benefits in turn are expected to contribute to improved product quality while reducing the time to first production.

3. Basic Concepts and Terminology

In order to specify the functionality of the RM, it is necessary to describe in a generic fashion the fundamental context in which it is designed to operate. The RM assumes the following model of a product development project:

- a standard set of processes, or tasks, exists with some associated ordering in terms of dependencies
- a set of agents, human and/or electronic, are assigned to execute these tasks
- these tasks specify a set of designs (i.e. for a product) characterized by components and/or features
- the project has a set of *requirements, representing goals or policies that should be met by* the designs under certain conditions of applicability
- for any given requirement, methods exist for ascertaining or predicting whether its conditions apply, and if so whether it is satisfied by the specified design.

A number of examples of requirements related to circuit board design are provided below. The conditions of these requirements generally relate to the state of the development process and the qualifying characteristics of the product. The requirements generally stipulate either desired objectives or mandated restrictions expressed in the form of quantitative or qualitative constraints.

For a typical ECA project, the number and complexity of requirements to be satisfied can be enormous, ranging from general project objectives (e.g. total cost must not exceed $4500) to specific design

constraints (e.g. mandatory in-circuit testing). Moreover, the responsibility for managing and meeting these requirements is usually distributed among different project participants, creating a communication challenge for the project leader or chief engineer.

Figure A-2 suggests a general hierarchical representation that we have adopted for organizing and describing requirements. The requirements are separated into a number of major categories (physical, mechanical, etc.), each of which expands partially into more detailed requirements. The RM system must not only track requirements and conflicts, but also support modification and extension of the requirements hierarchy, since each organization will have its own preferred view of how the representation should be structured.

There are generally three major sources of requirements that influence a development project and that may produce conflicts: organizational policies, manufacturing process constraints, and product objectives. These typically originate from different sources: Organizational policies are requirements, from both outside and inside the business, that govern the standards for buying materials and producing products (e.g. restriction to MIL-STD components). The sources are Design & Quality Assurance, Company Executives, Purchasing, Customers, Vendors, etc. Process constraints are requirements imposed by groups that have to build, test, support and maintain products (e.g., process constraints on surface-mount technology (SMT) might include detailed design rules, e.g. pin-to-pin spacing must be at least 25 mils). The sources include Reliability, Test, Manufacturing Engineering, Purchasing, Logistics, Component Engineering, etc. Product goals are requirements that come from the groups that have to design products (e.g. cost ceiling may be $5000 per unit). The sources are typically Customers, Marketing and Design.

4. Application to Product Development

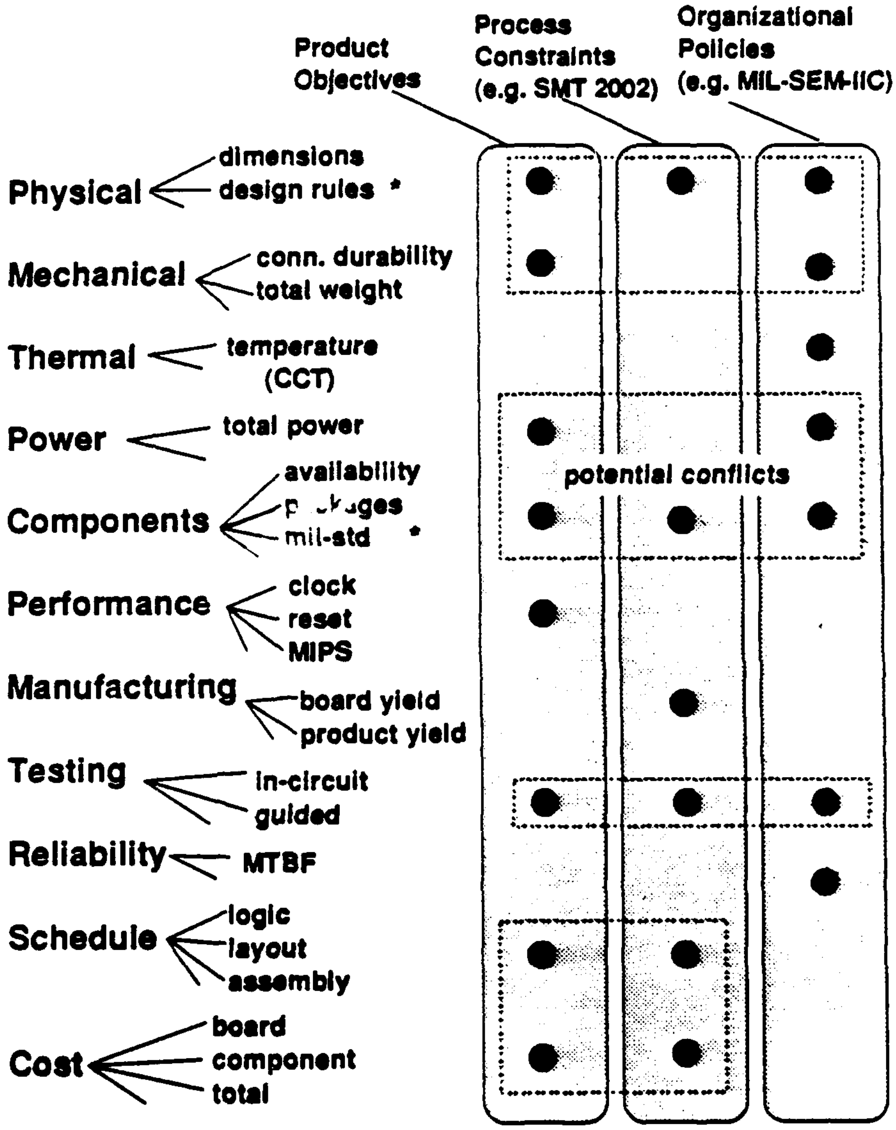
The RM typically will apply within a large engineering organization having multiple teams involved in different projects. We envision three classes of users:

- Project leaders perform an overall coordination function and monitor the status of the design vis-a-vis requirements. They routinely will specify and refine the overall project requirements.
- Individual engineers wish to understand the requirements relevant to their tasks or to evaluate the extent to which they are meeting requirements. They will occasionally modify requirements and introduce new constraints.
- System administrators support the other users by assisting in the definition and maintenance of requirement categories and associated evaluation methods.

The RM provides different human interface mechanisms to support each of these user groups. In addition, the RM provides a means for information sharing between multiple users, while allowing individuals to create their own private versions of requirements and evaluation results. The following sequence of actions suggests a typical usage scenario for the RM in a product development project:

- The project leader browses through an existing set of requirements and adds some new ones explicitly or by reference.
- An engineer views specific requirements (e.g. component costs) and tries to edit these but does not have the required level of authorization.
- An engineer checks compliance with cost requirements for the current parts-list extracted from the design file, noting violations and requesting detailed advice.

Figure A-2: Typical Sources of Requirements



* represent collections of requirements

- An engineer uses design tools, which automatically obtain requirements from RM, to search for alternative parts and substitute them into the design.
- An engineer requests reliability and testability evaluations for a newly completed design; the testability advisor provides advice on design enhancements causing the engineer to introduce new requirements.
- The project leader monitors overall progress against requirements, notes areas of shortfall, and considers alternative approaches.

5. Present status and future plans

Requirements Manager Version 1.0, an initial C-based implementation of the RM under Sunviews, has been demonstrated under the DICE program in 1989, and a number of extensions are planned. This year, we plan to move to the X11 Motif platform and to offer a C++ Application Programming Interface (API). Planned efforts for future extension include:

- enhanced capability to represent the full spectrum of requirements that may arise, including dependencies
- integration with the variety of external tools and databases needed to support evaluation
- enhanced management of requirements history, changes, and associated rationales
- an enhanced human interface that facilitates the introduction of new requirements and evaluation methods
- development of a domain-independent requirements system that can be customized easily to adapt to the key environmental factors within specific organizations.

As part of the continuing DICE program, we will be addressing these issues in the context of real DOD product development programs. This experience will provide a testbed for developing the existing RM prototype into a mature tool that can be utilized in a production environment.

DICE PROGRAM

Task 4.5.8.1 NDE/QA **GEAE**

Objective:

The objectives of this program are to develop and evaluate NDE inspection methods applicable to MMC structures and components and to establish controls and implement monitoring of the MMC process early in the manufacturing cycle. Conventional NDE inspection methods may not be adequate to fully assure the quality of MMC materials. This effort is needed to evaluate advanced NDE techniques and provide the methodology to effectively assess MMC core engine components and to assist in providing effective process quality assurance to reduce post process inspection requirements.

Approach:

Nondestructive Evaluation and Inspection methods are needed to assess the quality of Metal Matrix Composite sub-elements and sub-components and to develop and establish effective inspection methods during the component development and test work efforts. To accomplish this, inspection calibration and reference standards need to be defined and manufactured. These standards afford a means of measuring and reproducing the effectiveness and repeatability of the inspection method on a daily basis or over a period of time. Design of the calibration standards would be

based upon an Engineering definition of the type, size and nature of the defective condition(s).

In general, the defective conditions can be classified as 1) those related to the MMC structure, and 2) those related to bonded joints. Multiple standards of various designs are generally required to fully simulate the range and types of defects encountered.

Engineering had initially defined the following conditions as undesirable and in need of NDE detection methodology:

Incomplete consolidation - Delamination

Voiding/porosity

Fiber breakage/degradation/interface reaction

Incomplete or poor bonds

This list formed a basis for initial development considerations. Other undesirable conditions may exist but these would be covered at a later time in the overall Process Control and Quality Assurance Plan.

These reference standards and specimens would be initially evaluated with conventional NDE methods which would provide a screening test to identify the more obvious defect conditions. Identification of some of these defective conditions would require development and application of other more advanced NDE techniques

and methodologies. Selected specimens would be cut-up and examined by metallography to establish correlation between material condition and the NDE test results. From these results, the most promising NDE candidate method(s) for identifying the various defective conditions would be selected for potential scale-up and application to inspection of specific component assemblies, culminating in an inspection plan for production hardware.

Technical Results:

A suitable source to manufacture plasma spray specimens for the DICE program was not identified and the material for experimental measurements was not available for Phase II NDE work. Approximately 90% of the Phase II NDE development effort was contingent upon availability of this material. Therefore, only an extension of the Phase I effort toward establishing an advanced theoretical ultrasonic model and inspection of thirty CF6-80C2 LPT Stage 5 MMC blades was continued in Phase I.

Elements of the theoretical ultrasonic study reported in the Phase I final report covered 1) determination of the effective properties of the MMC using the individual component make-up of the MMC and their relative volume fractions, 2) introduction of the transformation and calculation of the properties for a laminate of any orientation from those of the reference values, 3) the outline of the derivation of the reflection and transmission

coefficients for normal incidence on multilayered plates immersed in liquid, and 4) determination of the different ultrasonic wave velocities as a function of the fiber volume fraction (V_f).

The Phase II theoretical ultrasonic study investigated effects of oblique incidence on defect characterization and plotted the predicted transmitted energy as a function of the F*D product (frequency x thickness) for various angles of sonic incidence. Various types of anomalies were built into the model and included varying degrees of porosity, fiber volume fraction, and matrix degradation.

Initial analysis clearly showed limitations of an inspection with the sound beam *normal or perpendicular* to the part surface to satisfactorily image the above mentioned defects. Experiments were designed to perform a pitch-catch type inspection with the transmitted sound beam and the received beam both at a 30 degree angle to the part surface. This configuration resulted in a means of locating the bunched fiber type defect in an 8 ply MMC panel. Dispersion curves for various fiber volume fraction values were plotted and showed significant differences in phase velocity at various FD products as the fiber V_f changed. However, this bulk measurement technique has an averaging effect which means that a 0.35 V_f sample can be distinguished from a 0.47 V_f sample, Figure 1, but one or two plys of 0.47 V_f in a 16 ply sample, where the other 14 plys are 0.35 V_f, will probably provide a measurement result that indicates the sample is 0.35 V_f, Figure 2.

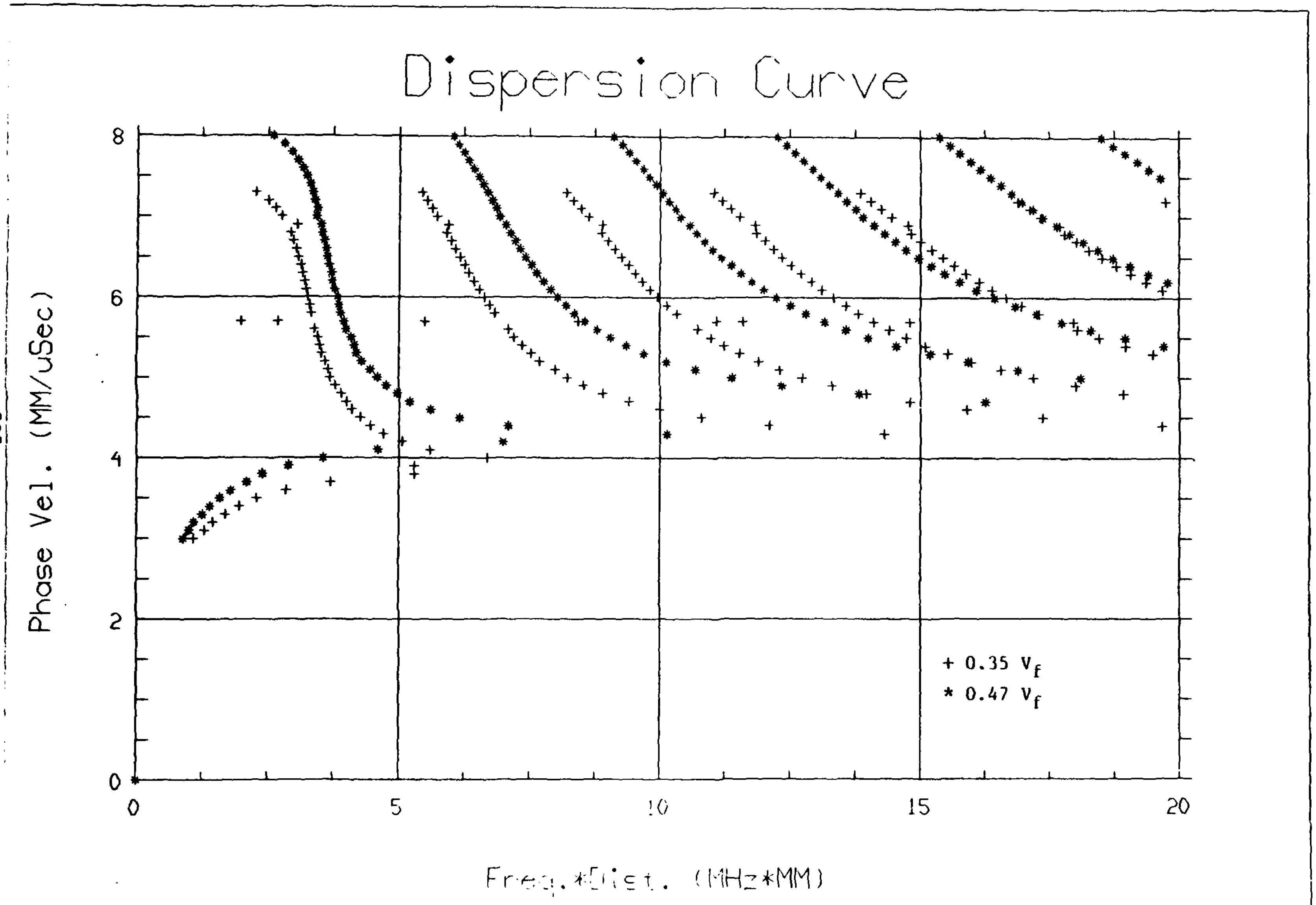

Dispersion Curve
Phase Vel. (MM/uSec)
Freq.*Dist. (MHz*MM)
0
2
4
6
8
0
5
10
15
20
+ 0.35 V_f
* 0.47 V_f

Figure 1

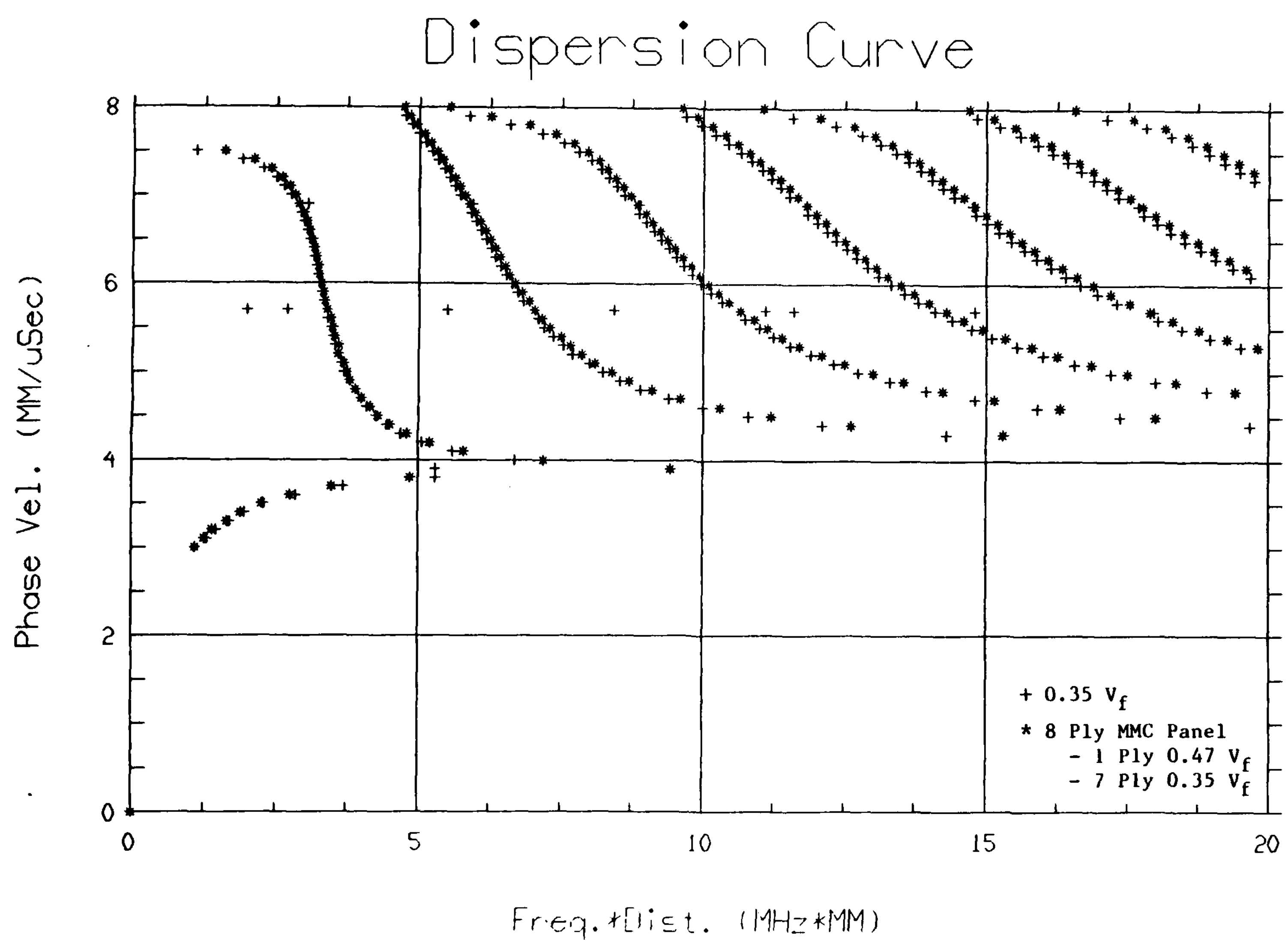

Dispersion Curve
Phase Vel. (MM/uSec)
Freq.*Dist. (MHz*MM)
0
2
4
6
8
0
5
10
15
20
+ 0.35 Vf
* 8 Ply MMC Panel
- 1 Ply 0.47 Vf
- 7 Ply 0.35 Vf

Figure 2

Thirty CF6-80C2 LPT Stage 5 (XD^{tm} matrix) MMC blades were inspected by microfocus x-ray and fluorescent penetrant NDE methods to determine the blade conditions before mechanical testing. Considerable porosity was shown in the dovetail area (at the "gate" area of the molding process) although some porosity was found in the airfoil section on several of the blades. A considerable amount of cracking was found in the dovetail area of the blades and was thought to be cracks from the dovetail grinding process.

Conclusion

To review the Task I NDE conclusions of the theoretical study, one can take any material (fiber & matrix) and obtain their equivalent composite properties from their individual properties. From these properties the different ultrasonic velocities in the material can be computed. Subsequently, the best transducer frequency for maximum transmission for a particular laminate construction can obtained. Transducer selection for the through-transmission (normal incidence) experiment was simplified to that showing best transmission. Data obtained from this analysis was useful in performing the through-transmission screening ultrasonic evaluation of MMC panels.

Task II conclusions are:

- The Titanium 6/4 matrix and SCS6 fiber MMC system is only mildly anisotropic. This can best be seen by comparing the wave speeds in the two perpendicular directions. Here, for a 0.35 fiber volume fraction specimen, these are given by 7.979 and 7.49 km/s along and normal to the fiber direction, respectively. This difference of about 6% is small compared with differences which exceed 85% for graphite-epoxy composites, for example.

- Due to this small anisotropy, it is more difficult to discriminate between situations which include defects such as ply mis-orientation, bunched fibers, and the like. Here, normal incidence sound wave transmission and reflection is not adequate to detect such anomalies. As we have demonstrated in this work, oblique incidence techniques have better potential in detecting such classes of defects.

- Delamination and interfacial cracking can be just as easily detected by normal incidence through-transmission inspection techniques.

- Material degradation due to distributed microcracks, porosity and the like can be detected by both normal and oblique incidence techniques. Oblique incidence is a more viable technique, however.

Recommendations

At this time no NDE is planned by the project for Phase III although it is still vitally important to complete the experimental work. If it appears that a MMC material supplier can be agreed upon and the required samples fabricated, NDE activity should again be initiated.

DICE Program

Task 4.5.8.2 Quality Advisor **West Virginia University**

Objectives:

The main objective of this task was to provide timely advice on quality assurance to designers in a concurrent engineering environment and to ensure that quality considerations were incorporated into all stages of a product life cycle. The goal of this task was to provide professional expertise and computerized assistance on various aspects of quality function to designers, prototype builders and production personnel.

The other objective of this task was to explore and incorporate traditional design of experiments (DOE) and product parameteroptimization methods which can be effectively used in designing products that are less sensitive to sources of variability which may adversely affect their performance and reliability.

Approach:

To achieve the above-mentioned objectives, the following steps were taken:

1. Development of Quality Assurance Advisor (QAA). One of the main vehicles for the delivery of advice on quality is the Quality Assurance Advisor. This advisor must be easy to use, comprehensive and portable. Product designers, who are not intimately familiar with statistical concepts of planning experiments and methods of data collection and analysis,must be able to make use of this advisor with minimal or no help from a human expert. The advisor must also contain methods of process optimization and control to cover the prototyping and production stages;

2. Development and application of parameter optimization techniques. These methods are useful when the product is at the design stage and it is not possible to design and conduct experiments and collect data to determine the statistical relationship between the performance measure (dependent variable) and various product parameters (independent variables) and

3. Theoretical Developments. To advance the understanding of newer approaches to quality assurance, this task embarked on conducting research on consumer behavior and assessment of quality under various risk conditions.

Technical Results:

Work was performed on both off-line and on-line quality assurance procedures. Off-line procedures were used at the product and process design stages such that quality related problems can be anticipated and dealt with at the design stage. On-line quality assurance methods were carried out after the product design was complete. They included the use of statistical process control (SPC) methods and Taguchi's on-line quality assurance procedures to ensure that design requirements were met economically at the prototyping and production stages.

A Quality Assurance Advisor (QAA) was developed to incorporate both on-line and off-line procedures. As an advisor, the QAA makes it possible for product designers to screen a large number of product parameters through designed experiments, analyze the results and find the combination of parameters that result in optimal performance.

The framework for the QAA was designed for workstations such as the Sun4, VAX 3200, and Silicone Graphics. The user interface of the QAA was developed using the library tools of the "Object-Oriented User Interface Builder" (OUIB) developed at WVU. This interface builder is compatible with X-11 Window and DEC-Window systems.

The QAA can be used as a stand alone capability or as part of the DICE network. Interaction with the QAA may take place in two ways:

a. Information about product design parameters may be generated through statistically designed experiments at the prototyping or manufacturing stages. Data is then fed into the off-line and on-line modules of the QAA and

b. Information may be generated via software that take advantage of known physical and mathematical relationships between design parameters and product performance measures such as finite element analysis methods. This information is then fed into the optimization modules of the QAA to find the optimum combination of parameter settings.

At present, the QAA includes four main menu items: on-line procedures, off-line procedures, help and utilities. A description of the on-line and off-line procedures is given below.

1. Off-line Quality Assurance: The main tools developed include experimental design and optimization techniques. Included below is a brief description of the main off-line modules available and how they can be used in the design process.

   - Fractional Factorial Module - This module generates two level full or fractional factorial designs. It can generate a full factorial design with up to 8 factors or a fractional factorial design with up to 12 factors and a fractionalization element as high as 8. For the fractional factorial designs, the module generates a standard defining contrast. Alternatively, the user can enter his own defining contrast;

   - Yates Module - This module uses the Yates algorithm to perform analysis of variance on the data collected for full or fractional factorial designs at 2 levels;

   - ANOVA Module - This module performs the analysis of variance on balanced designs with up to 4 factors, each with up to 8 levels. For a larger number of factors, the number of levels is limited to 4;

   - Regression Module - Regression analysis is used as a tool to estimate the value of a response variable as a function of some independent variables. It may also be used to investigate the relationship between the response variable and several independent variables and their interactions, as defined mainly from the results of the Yates and the ANOVA modules, and

   - Optimization Module - This module may be used to find the values of the independent variables that lead to the optimization of the response variable. It may also be used as a stand alone optimization program. The module uses a combination of a modified "steepest ascent" search procedure based on EVOP and a univariate walk to search for the optimal point on the response surface.

2. On-Line Quality Assurance: A brief description of the on-line quality control modules included in the QAA is given below:

- Mean Squared Drift - This module computes the mean squared drift which is an important measure of the deviation of the process from its target value.

- Process Parameter Tolerance - This module is used to determine the process tolerance, i.e., how far the process can be allowed to deviate from its optimal setting before it starts to produce nonconforming products.

- Feedback Product Quality for Variable Characteristics - Given appropriate information on the various process parameters and costs, the program computes the optimal process adjustment limit the optimal adjustment interval and the total quality cost;

- Process Parameter Control for Variable Characteristics - This is similar to the previous module with the exception that it considers the process parameters rather than the product characteristic and its specifications;

- Process Control for Attribute Characteristics - Previous modules were concerned with variable characteristics, i.e., those that can be measured. This module deals with attribute characteristics that result from categorization of products. This module contains the routines for optimal process control for such characteristics, and

- Control Methods for Process Improvement - Parameters that affect the total cost of operating a production system can be categorized into three categories. They are: (1) parameters associated with the production process; (2) parameters related to diagnostic steps and (3) parameters related to recovery and adjustment of the production process. This module can be used to improve these parameters and to reduce the total loss function.

Besides the development of the QAA, theoretical work was started to investigate consumer behavior under quality risk. Consumers experience a loss due to product quality variance. While the price paid by all consumers of a given item maybe the

same, each will experience a different amount of loss due to variations in product quality. A model is currently being developed to describe consumer choice of a product with multiple quality attributes. Consumers usually assign different weights to independent quality attributes in addition to the price of the item. The amount of loss experienced by each consumer is a function of quality deterioration for each attribute and the individual weight assigned to that attribute. A consumer quality function needs to be derived such that consumer budget, product price, quality variance and information uncertainty are taken into consideration.

Conclusions:

Work was performed on both off-line and on-line quality assurance procedures. A "Quality Assurance Advisor" (QAA) was developed to help incorporate quality into a concurrent engineering environment both at the product design and manufacturing stages. The QAA consists mainly of off-line modules that can be used at the design stage and on-line modules that can be used at the production stage. These modules cover classical quality control procedures as well as recent developments in quality assurance that include such tools as design of experiments, optimization and Taguchi techniques. Additionally, the QAA was designed to provide capabilities for in-house utilities and "help" instructions.

The QAA is completely self-contained. It assembles important quality assurance tools in one user-friendly package that enables users who are not experts in quality control to find and apply the procedures needed in their quest for improved quality. The QAA was also designed to provide the necessary help needed in using these procedures. As a result, the QAA aids in the reduction of time needed to design, prototype and produce a quality product and aids in maintaining this quality during its production.

Theoretical work also began in the investigation of consumer behavior under quality risk. Detailed results and conclusions from this work will be published in the future.

Recommendations:

The QAA provides valuable "quality assurance" help and tools for designers in a concurrent engineering environment. It is easy to use, comprehensive and portable. Its users need not be intimately familiar with statistical concepts of design of experiments and quality control. The QAA can be enhanced by the inclusion of more

modules or expansion of its existing modules and by completion of its "help" and "utilities" capabilities.

Besides the QAA, standard statistical software (e.g., SAS) and commercial packages that carry out Taguchi's techniques should be made available to provide users who are familiar with statistics and quality control procedures with more sophisticated tools. The optimization module, which is included as part of the QAA, can be used as a stand alone tool and can be used in numerous ways. This module can be enhanced by adding capabilities such as "constrained optimization."

Publications:

1. Jaraiedi, M., Iskander, W., "A Quality Assurance Advisor for a Concurrent Engineering Environment." Proceedings: Second National Symposium on Concurrent Engineering, Morgantown, February, 1990.
2. Jaraiedi, M., Iskander, W., "Contributions of Quality Control in a Concurrent Engineering Environment." Proceedings: Annual International Industrial Ergonomics and Safety Conference, Cincinnati, June, 1989.

Hardware:

None.

DICE PROGRAM

Task 4.5.10 Manufacturing Demo-Fan Blade GEAE

Objectives:

This program was initiated to demonstrate the Ti6-4/SCS6 metal matrix composite portion of the overall DICE effort to explore the FOD capability of Graphite/Epoxy (GR/Ep) and Titanium/SCS6 (MMC) composites for Ultra Bypass Engine (UBE) blades. The objectives were to:

1. Explore the bird strike capability of both systems via progressive refinement of manufacturing options within the design envelope and select one system.
2. Concurrently explore options for minimizing the weight of composite blades.
3. Establish the computer architecture and software to model the manufacturing and design options necessary to optimize FOD resistance of both systems while meeting all other blade requirements.

Approach:

The approach used for the Ti 6-4/MMC portion of the program was to design and fabricate MMC birdstrike test sections within the same overall envelop as the Graphite/Epoxy section. The test section, a slab sided configuration that simulates a constant chord section of the UBE blade without twist or camber, was designed so that three different MMC placements could be evaluated. That is, longitudinal MMC placement extending all of the way to the leading edge and two different distances (2 and 4 inches) from the leading edge. Drawing 4013401-533 illustrates the key features of this section as shown in figure 1. Four blade sections for each MMC placement were to be made.

Textron Specialty Materials, Lowell, MA, was the fabrication vendor. The fabrication method chosen was the fiber/foil process in which alternate layers of Ti 6-4 foil and SCS6 filament fabric (cross-woven using Ti 6-4 ribbon) were cut to the required stepped sizes for the MMC inserts and laid up into machined cavities in the central Ti 6-4 monolithic core. A 14 mil Ti 6-4 cover sheet was then fitted over the entire blade/core surface, Electron Beam welded into place, evacuated, leak checked, and the assembly HIP'd to consolidate and bond the MMC to the central blade core. One trial blade section was to be fabricated and evaluated before the balance of blades were fabricated to enable changes to be made if needed.

Following fabrication and NDE the sections were to be evaluated in bird strike tests in parallel with similar Gr./Epoxy sections. System selection, either Ti 6-4/MMC or Gr./Epoxy, was to be made after post test evaluation and that system carried forward through complete fan blade development.

Technical Results

Although the original plan for this program called for fabrication and evaluation of a trial blade prior to fabricating the remaining 12 blades, the actual timing of the program did not allow this trial blade to be made in advance of the others. Thus, all 13 blades were processed as a group.

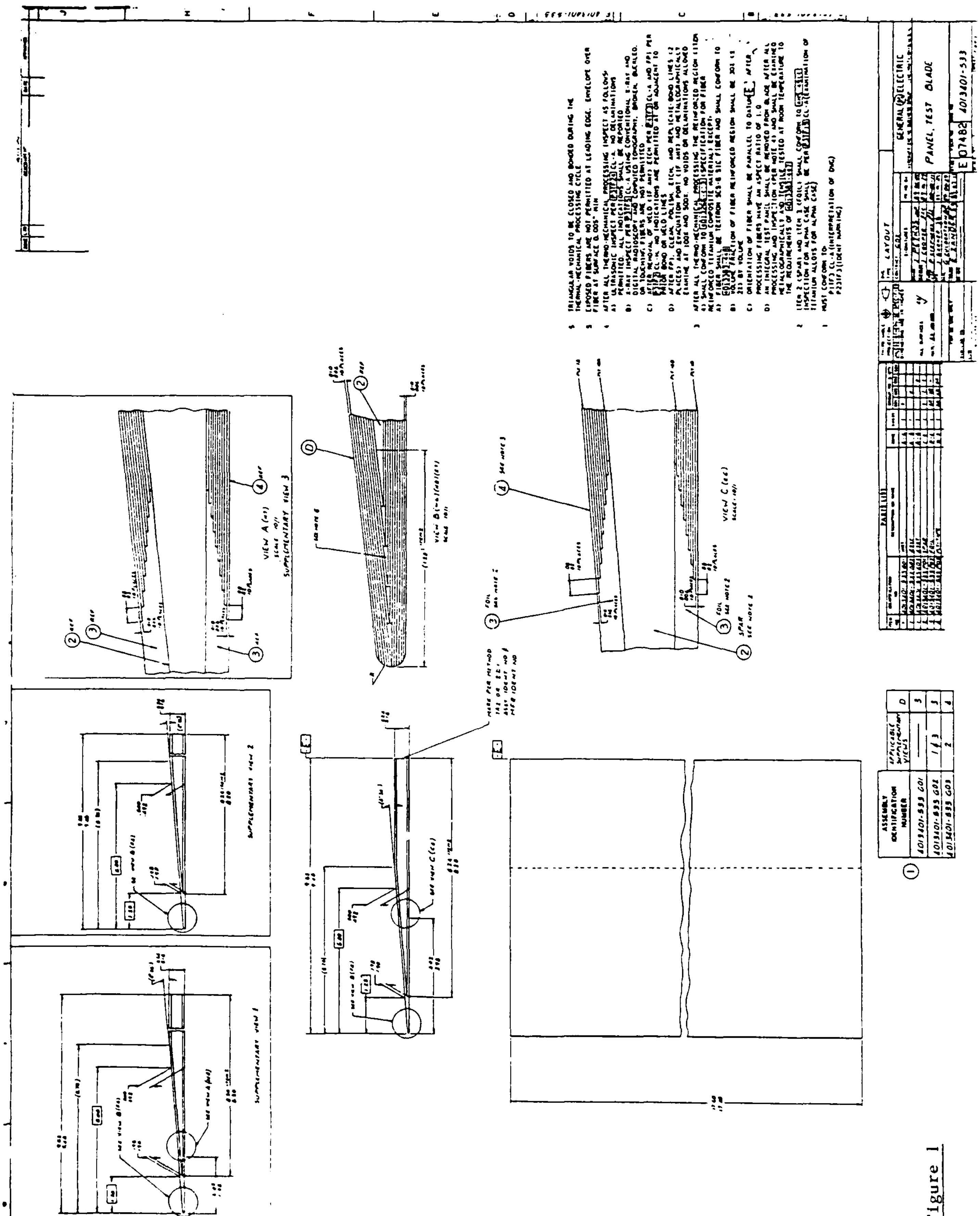


Figure 1

The overall fabrication/inspection process flow chart for blade manufacture is shown in figure 2. A typical example of a manufacturing operation sheet showing the construction of the blade with MMC reinforcement extending to the leading edge is shown in figure 3. In process assembly steps for this blade are illustrated in figure 4 which shows the lower most ply (1B) at the top of the figure. The middle picture shows the 10 lower plies (1B through 10B) and the Ti 6-4 machined monolithic core, while the bottom picture shows the lay up at the 2nd ply above the Ti 6-4 core (9A ply). The sketch in figure 5 illustrates the blade with the Ti 6-4 cover sheet EB welded in place and shows the TIG welded evacuation tube.

A visual examination of the 13 blades after HIP consolidation at 1650F/15ksi/1hr. indicated that the trial blade lost vacuum and did not consolidate. None of the remaining twelve had evidence of failure. C-scan and X-ray examination revealed good bonding and fiber alignment. The eight blades partially reinforced with MMC (2 in. and 4 in. removed from leading edge) were rough machined to final dimensions by water jet cutting followed by a finish machining operation and dimensionally inspected. The four blades with MMC reinforcement extending to the leading edge exhibited some bowing at the leading edge which could be improved by a creep straightening operation, however, time constraints did not permit this operation.

Photographs showing one of each configuration are presented in figure 6. The MMC regions of the blades have a rough cosmetic appearance where the fiber/foil pack consolidated during HIP. This could be eliminated in future blades by using thicker cover foils which would permit a finish machining operation to improve the blade surface finish.

Also a depression is evident where the outgas tubes collapsed during HIP consolidation. For future blades this could be reduced or eliminated by relocating the tubes so that they would not interfere directly with the MMC areas.

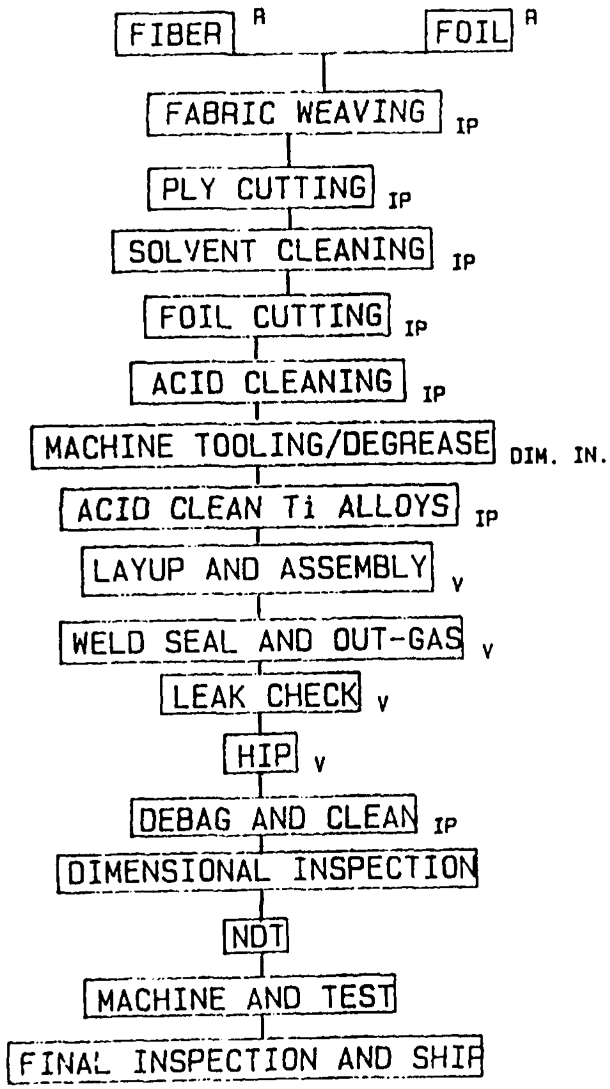
Conclusions:

1) Blade sections incorporating MMC inserts were successfully laid up, evacuated, and HIP consolidated using a process that required no external tooling or encapsulation material. The blades were completely self contained and as such, following consolidation, there was no need for mechanical decanning nor acid etching to remove external steel canning as with other processes.

2) The disadvantage of this fabrication method is that there was some distortion of the leading edges and a rough cosmetic appearance on the MMC surfaces since there were no external stiffeners, steel encapsulation sheets or hard tooling surfaces in contact with the MMC during bonding.

3) The program was discontinued before the Ti 6-4/MMC blade sections could be completely finished and tested.

FIGURE 2 . FABRICATION/INSPECTION PROCESS FLOW



R = RECEIVING INSPECTION
IP = IN-PROCESS
V = VERIFICATION
DIM. IN. = DIMENSIONAL INSPECTION

Figure 3: MMC Extending to Leading Edge

MANUFACTURING OPERATION SHEET

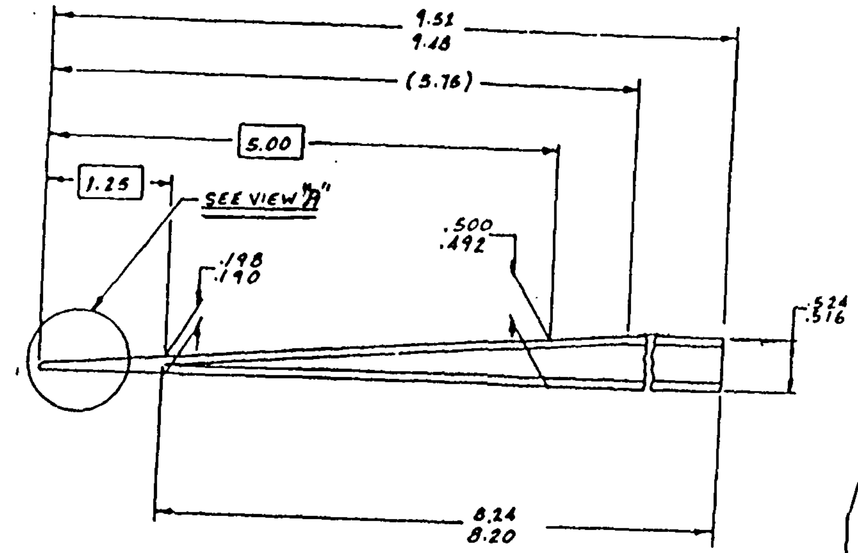
OPERATION NUMBER: 0040

LOT NUMBER:

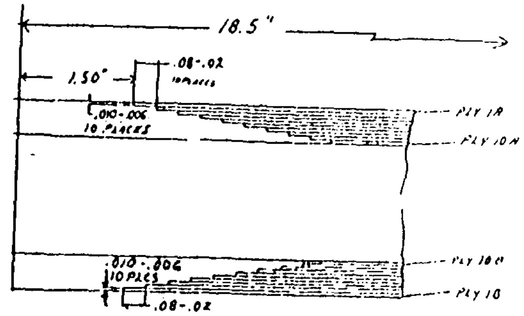
OPERATION REV:

SHEET

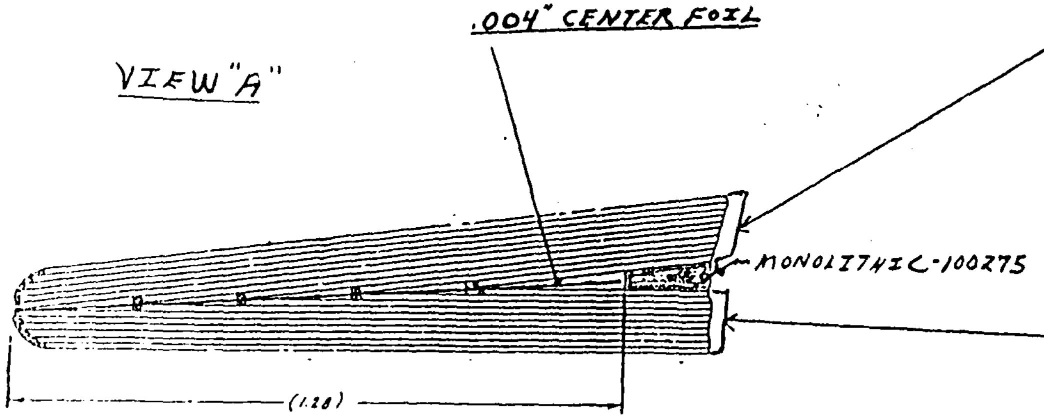
FIGURE 40-1



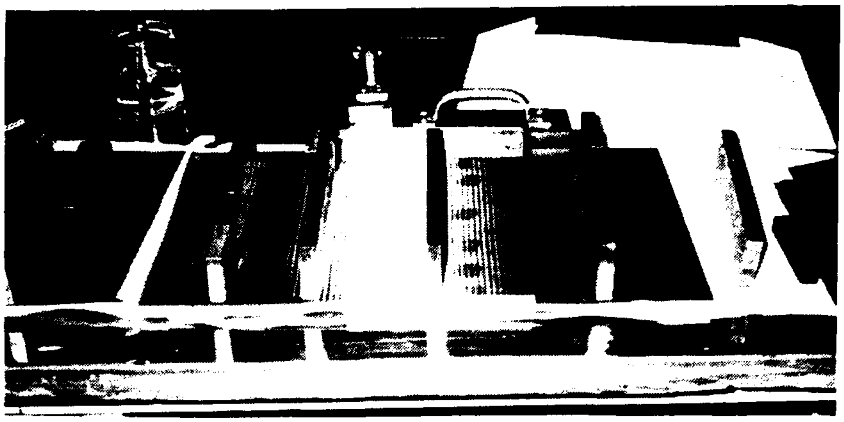
SECTION VIEW OF BLADE ALONG 18.5" LENGTH - SHOWING TAPERED EDGES.



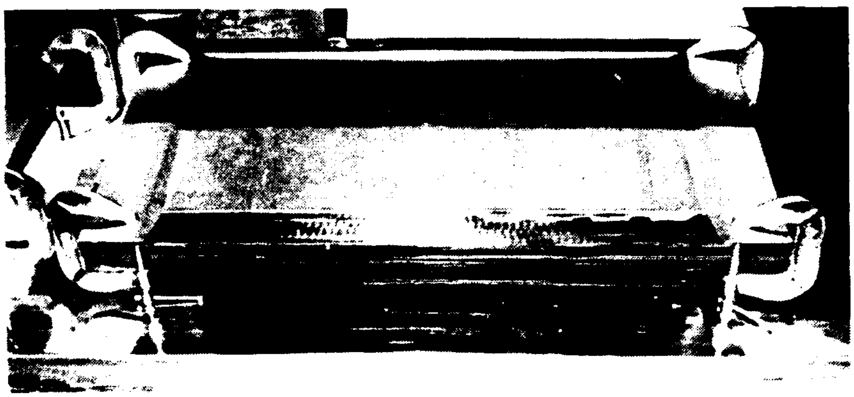
VIEW "A"



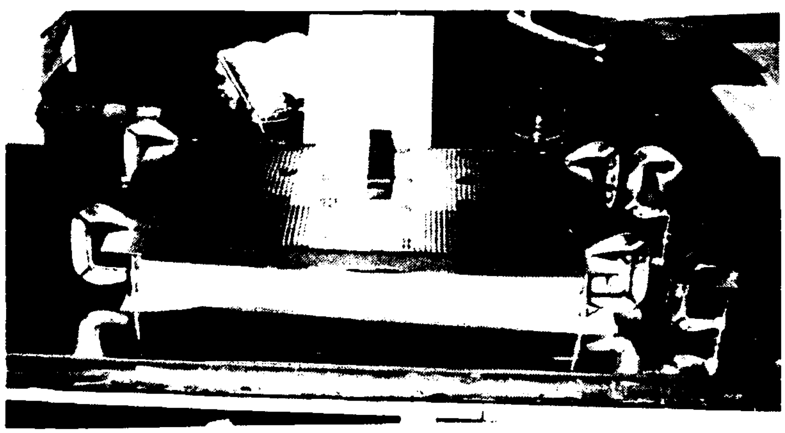
FOIL NO.		PLY NO.
1	.004	1A
2	.004	2A
3	.004	3A
4	.004	4A
5	.004	5A
6	.004	6A
7	.004	7A
8	.004	8A
9	.004	9A
10-1	.004	10A
10-2	.004	
C.F.	.004	16.5" x 1.28"
10-2	.004	10B
10-1	.004	9B
9	.004	8B
8	.004	7B
7	.004	6B
6	.004	5B
5	.004	4B
4	.004	3B
3	.004	2B
2	.004	1B
1	.004	



Beginning - Lower Most Ply (1B)



Halfway - Lower 10 plies (1B to 10B) plus machined monolithic Ti6-4 core.



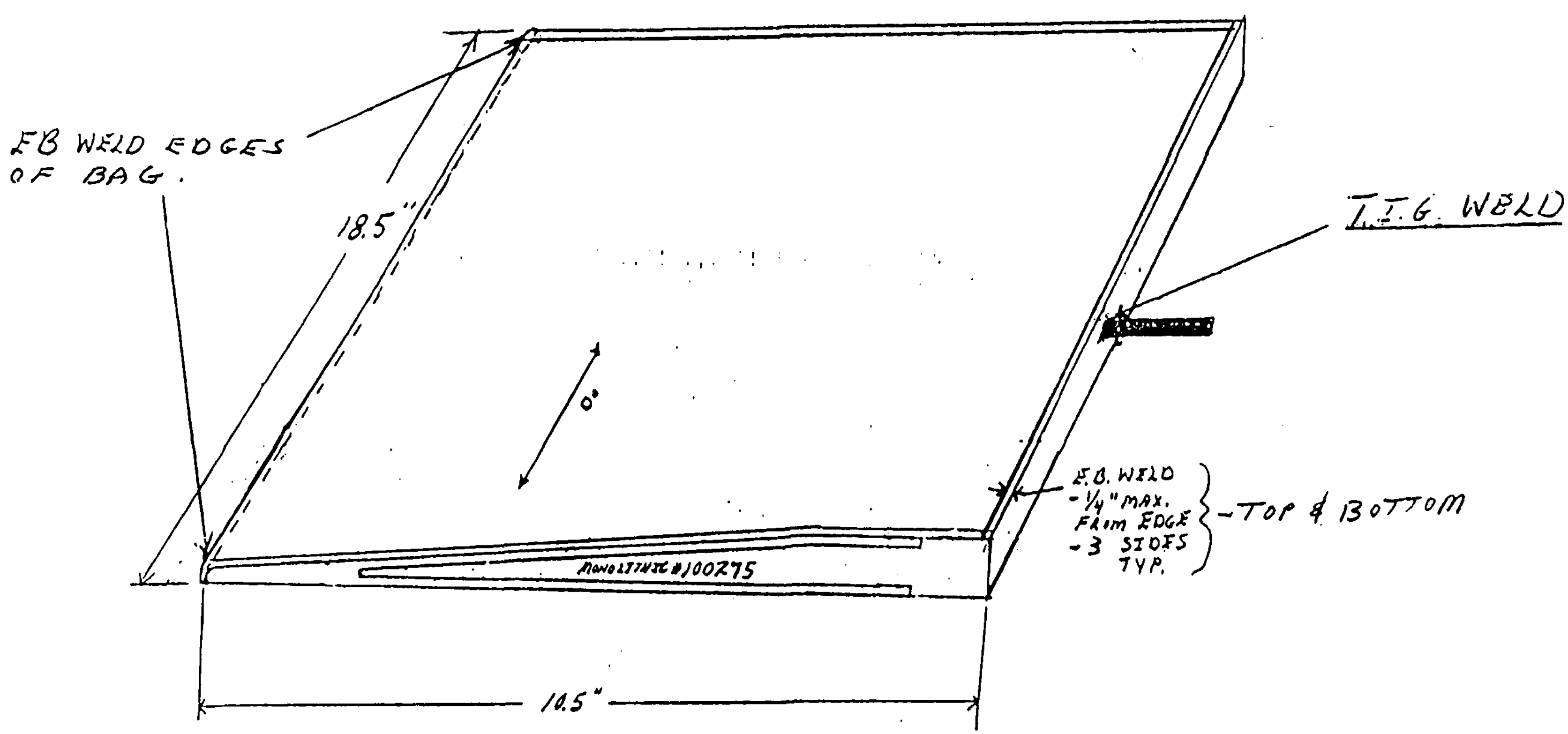
Top Plies - 2nd ply (9A) above machined monolithic Ti6-4 core.

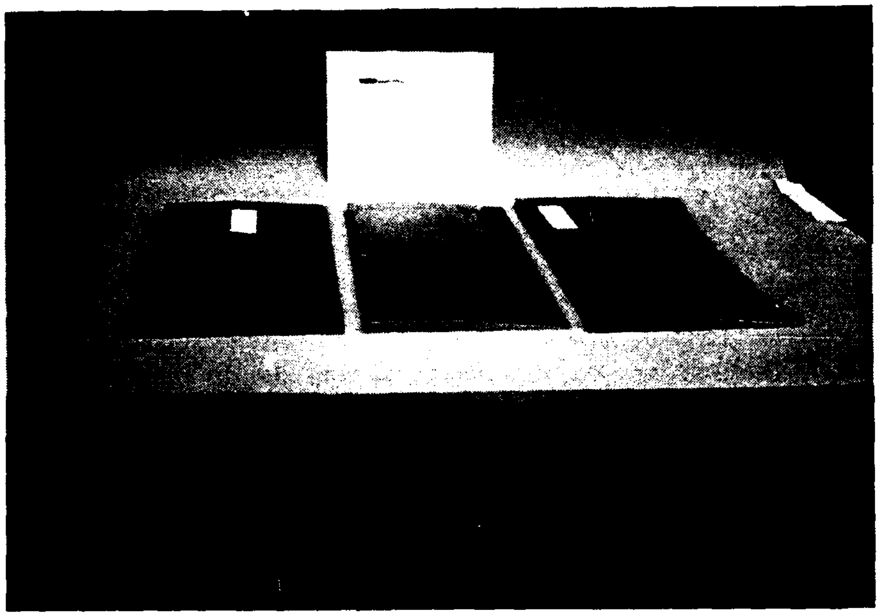
Figure 4. In-process assembly steps for blade section with MMC extending to leading edge.

Figure 5: MMC Extending To Leading Edge

MANUFACTURING OPERATION SHEET		OPERATION NUMBER: 0050
LOT NUMBER:	OPERATION REV:	SHEET

FIGURE 50-1





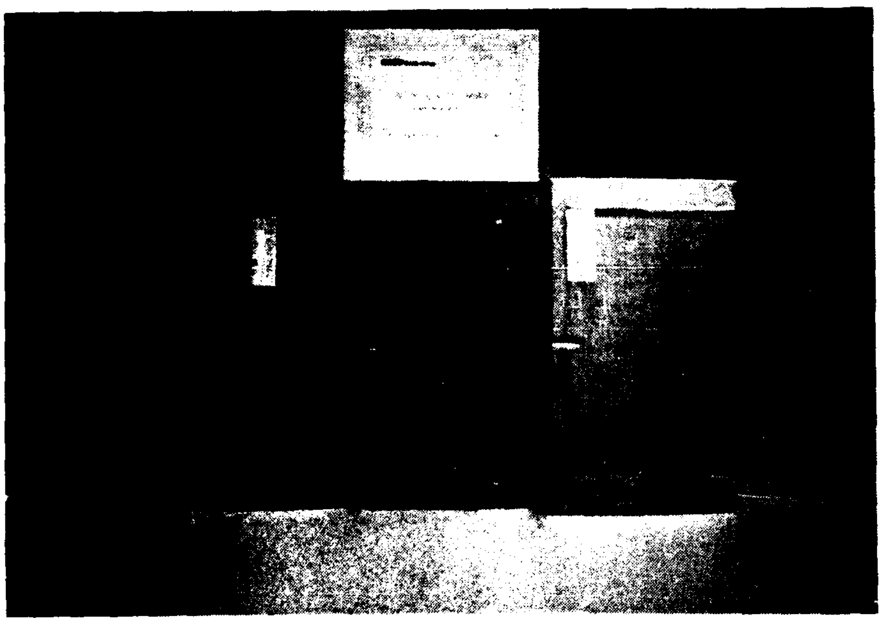


Figure 6: Ti6-4/MMC wide chord fan blade demo sections.

- Right - MMC extends to leading edge
- Middle - MMC extends to 2" of leading edge
- Left - MMC extends to 4" of leading edge

Recommendation:

1) The Ti 6-4/MMC blade sections should be completed through manufacture and subjected to bird strike tests in order to complete the back to back evaluation against the Graphite/Epoxy blades.

2) The bowing distortion deficiencies can be corrected through creep straightening operations at elevated temperatures. This corrective action has been successfully demonstrated on MMC panels and other components and should be done on these parts.

3) Corrective action for surface roughness should be demonstrated by consolidation of additional blade samples with heavier cover sheets that will allow finish machining to a smooth surface.

Publications: None

Hardware:

The following Ti 6-4/MMC blade sections are in inventory.

1) 4 blade sections with MMC placement extending two inches from the leading edges.
2) 4 blade sections with MMC placement extending four inches from the leading edge.
3) 4 blade sections with MMC placement extending all the way from the leading edge, but not straightened and the edges unmachined.

BIBLIOGRAPHY OF DICE
Publications

4.0 Bibliography of DICE Publications

1. "Observation of a Localized Moment in Mn Doped γ-TiAl Alloys", J. Coletti, (Supervised by M.S. Seehra), Masters Thesis, West Virginia University, 1990.

2. "Life-Cycle Features for Computer-Assisted Design", Scott A. Safier and Susan Finger, to be presented at the International Conference on Engineering Design, ICED, Dubrovnik, August, 1990.

3. "Parsing Features in Solid Geometric Models", Scott A. Safier and Susan Finger, to be presented at the European Conference on Artificial Intelligence-90.

4. "Representing and Recognizing Features in Mechanical Designs", Susan Finger and Scott A. Safier, to be presented at the *Second Annual International Conference on Design Theory and Methodology*, Chicago, September, 1990.

5. "Interval Approaches for Concurrent Evaluation of Design Constraints", D. Navinchandra and J.R. Rinderle, *Concurrent Product and Process Design,* ASME, DE-Vol. 21, PED-Vol.36, December, 1989.

6. "A Blackboard Scheme for Cooperative Problem-Solving by Human Experts", F. Londono, K.J. Cleetus, and Y.V. Reddy, MIT Conference on Cooperative Product Development, Morgantown, WV, November, 1989.

7. "A Blackboard Problem-Solving Model to Support Product Development", F. Londono, K.J. Cleetus, and Y.V. Reddy, *Second National Symposium on Concurrent Engineering*, Morgantown, WV,February, 1990.

8. "The Local Concurrency Manager in Distributed Computing", R. Raman, K.J. Cleetus, and Y.V. Reddy, *Second National Symposium on Concurrent Engineering*, Morgantown, WV, February, 1990.

9. "DICE Network Monitoring", James M. Coleman, and William H. Dodrill, *Second National Symposium on Concurrent Engineering*, Morgantown, WV, February, 1990.

10. "Using a Relational Database as a Index to an Object Oriented Database in Design Applications", M. Hardwick, and G. Samaras, *Second International Conference on Data and Knowledge Systems for Manufacturing and Engineering*, at NIST in Gaithersburg, October, 1989.

11. "ROSE: A Database System for Concurrent Engineering Applications", M. Hardwick, D. Spooner, E. Hvannberg, B. Downie, J.A. Faulstich, D. Loffredo, A. Mehta, and D. Sanderson, *Second Annual Conference on Concurrent Engineering*, West Virginia University, February, 1990.

12. "The Evolution of ROSE: An Engineering Object-Oriented Database System", D. Spooner, M. Hardwick, E. Hvannberg, D. Downie, D. Loffredo, A. Mehta, J.A. Faulstich, D. Sanderson, R. Harris, G. Abou-Ezzi, J. Gong, J. Young, M. Rovira, and G. Samaras, *Proceedings of the Second Rensselaer International Conference on CIM*, May, 1990.

13. "A Data Translator for Engineering Systems", D. Spooner, D. Sanderson and C. Charalambous, *Second International Conference on Data and Knowledge Systems for Manufacturing and Engineering*, Gaithersburg, MD, October, 1989.

14. "Electronic Design Notebook for the DARPA Initiative in Concurrent Engineering", W.H. Uejio, *Proceedings of the Second National Symposium on Concurrent Engineering*, in February, 1990.

15. "Encoding Logical Constraints into Neural Network Cost Functions", D. Thomae, and D.E. Van den Bout, to appear in the *Proceedings of the IEEE International Conference on Neural Networks*, 1990.

16. "The Knowledge Server Module of the DICE System: Phase II", Dennis H. Wong, Pedor Oscar Cubillos, Ravi S. Raman, Michael Bender, David Dymm, *Proceedings of the Second National Symposium on Concurrent Engineering*, Morgantown, WV, February, 1990.

17. "Concurrent Design", S. Finger, et al., special issue of *Applied Artificial Intelligence.*

18. "The Role of Architecture in Computer-Assisted Design Systems", Scott A. Safier, Mark S. Fox, January 30, 1990.

19. "Machining Advisor in Concurrent Engineering", B. Gopalakrishnan, *Proceedings of the IIE Integrated Systems Conference*, Atlanta, 1989.

20. "Computer Aided Simulation of H Section Forging Using ALPID", S.N. Dwivedi, submitted to *ASME Journal of Engineering for Industry.*

21. "Development of Constitutive Equations and Processing Maps of Inconel 718", S.N. Dwivedi, submitted in *ASM Journal for Material and Processing.*

22. "Computer Aided Simulation of H Section Forging using ALPID", S.N. Dwivedi, and R. Shankar, *ASME Conference on Intelligent Processing of Materials*, 1990.

23. "Enhanced Forging through Computer Simulation", S.N. Dwivedi, R. Shankar, *Journal of Advanced Materials and Processes*, pp. 23-31, February, 1990.

24. "Computer Aided Design for Manufacture of Rolled Rings", R. Shankar, and S.N. Dwivedi, accepted for publication in *International Journal of Computer Integrated Manufacturing*.

25. "A General Framework for Local Interpolation", C.K. Chui and H. Diamond, TAMU CAT Tech. Report, June, 1989.

26. "Fundamental Splines from Spline Spaces", CERC Tech. Report, March, 1990.

27. "Polar-Spline Approximation", C.Q. Zhang, Technical Report TR88-10, Department of Statistics and Computer Science, West Virginia University; ARCHIVE-0023-89, Concurrent Engineering Research Center.

28. "Error Analysis of Polar-Spline Approximation", C.Q. Zhang, Technical Report TR89-2, Department of Statistics and Computer Science, West Virginia University.

29 "Polar-Splines and Offsets", C.Q. Zhang, Technical Report TR89-3, Department of Statistics and Computer Science, West Virginia University; ARCHIVE-0022-89, Concurrent Engineering Research Center.

30. "Existence of Offset", C.Q. Zhang, Technical Report ARCHIVE-0019-89, Concurrent Engineering Research Center.

31. "Proper Offsets of Closed Convex Curves", C.Q. Zhang, Technical Report ARCHIVE-0024-89, Concurrent Engineering Research Center.

32. "Intersections of Curves and Existences of Proper Offsets", C.Q.Zhang, Technical Report ARCHIVE-0025-89, Concurrent Engineering Research Center.

33. "Offsets of Curves", C.Q. Zhang, Technical Report ARCHIVE-0026-89, Concurrent Engineering Research Center.

34. "Parametric Design of Laminated Composite Flange", B.S. Kang, J. Prucz, and F. Hsieh, to be presented in Fifth International Conference on CAD/CAM, Robotics and Factories of the Future, Norfolk,VA, December 2-5, 1990.

35. "Parametric Finite Element Modeling Techniques in Concurrent Engineering Environments", V.H. Mucino, J.E. Sneckenberger, J.C. Benner, and S. Chung, *Proceedings of the Second National Symposium on Concurrent Engineering*, February 7-9, 1990.

36. "A Practical Engineering Approach for Predicting Interlaminar Stresses in Composites", J. Prucz, and M. Lambi, to be presented in Fifth International Conference on CAD/CAM, Robotics and Factories of the Future, Norfolk, VA, December 2-5, 1990.

37. "Parametric Modeling and Analysis of Turbine Blades for Concurrent Engineering Environments", M.K. Ruth, Masters Thesis, Department of Mechanical and Aerospace Engineering, West Virginia University, May 1989.

38. "Parametric Design of Composite Ducts and Pressure Vessels, J. Sivan, J. Prucz, and P.C. Upadhyay, to be presented at ASME Winter Annual Meeting, Dallas, TX, November 25-30, 1990.

39. "Coordination of Requirements Management in Printed Circuit Board Design and Production", J. Fiksel, F. Hayes-Roth, *Second National Symposium on Concurrent Engineering,* Morgantown, WV, February, 1990.

40. "Concurrent Design for Testability", A. Matsumoto, V. Jagannathan, V. Saks, C. Buenzli, Workshop on Concurrent Engineering Design, International Joint Conference on Artificial Intelligence (IJCAI), Detroit, MI, August 1989.

41. "Concurrent Design for Testability of Hierarchical Boards", A. Matsumoto, V. Jagannathan, V. Saks, C. Buenzli, *Second National Symposium on Concurrent Engineering*, Morgantown, WV, February, 1990.

42. "A Rule Based Expert System for Design and Assembly of Printed Circuit Boards", S.K. Padhy, and S.N. Dwivedi, Fourth International Conference on CAD/CAM, Robotics and Factories of the Future, December, 1990.

43. "New Approaches to Computer-Integrated Manufacturing", M.R. Fehling, Special Colloquium presented at the Systems Research Center, University of Maryland, College Park, Maryland, March, 1990.

44. "The Heuristic Control Virtual Machine: An Instance of the Schemer Architecture for Real-Time Problem-Solving", M.R. Fehling, A. Altman, and B.M. Wilber, in V. Jagannathan, R. Dodhiawala, and L. Baum (Eds.) *Blackboard Architectures and Applications*, New York, Academic Press, 1989.

45. "Concurrent R&D and the Product Design Process", E. Tse, *Laboratory for Intelligent Systems Technical Report No. ISL-TR-90-11*, February, 1990.

46. "Encoding Logical Constraints into Neural Network Cost Functions", D. Thomae, and D.E. Van den Bout, to appear in the *Proceedings of the IEEE International Conference on Neural Networks*, 1990.

47. "A Quality Assurance Advisor for a Concurrent Engineering Environment", M. Jaraiedi, W. Iskander, *Proceedings: Second National Symposium on Concurrent Engineering*, Morgantown, WV, February, 1990.

48. "Contributions of Quality Control in a Concurrent Engineering Environment", M. Jaraiedi, W. Iskander, *Proceedings: Annual International Industrial Ergonomics and Safety Conference*, Cincinnati, OH, June, 1989.